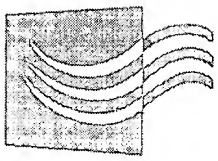
Instance Monitor™

User's Guide

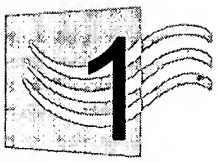
Version 1.0

Quest Software
610 Newport Center Drive, Suite 1400
Newport Beach, CA 92660
USA
Tel. (949) 720-1434 / Fax. (949) 720-0426
support@quests.com

Instance Monitor
User's Guide
Updated – May, 1999
Software Version – 1.0



Contents



Introduction

Instance Monitor is a powerful database monitoring and diagnostic tool. Its unique user interface provides you with an intuitive, visual representation of the activity on your database. Graphical flows illustrate the rate at which data is moving between database components. Icons display the value of key statistics and metrics.

The power of Instance Monitor lies in its ability to provide visual and audible warnings if the performance metrics exceed acceptable thresholds. The components and dataflows change color to show you the source of the problem.

A range of reports and graphs provide you with detailed information about your database. This information can be viewed on the screen or printed.

You can set Instance Monitor to warn you when a threshold is reached. You may set a number of thresholds so that warning messages are displayed well before the traffic levels into or out of a database become critical. Instance Monitor uses a number of different techniques to warn you when your database is exceeding a threshold.

This Guide contains the information you require to interpret the data Instance Monitor is providing. You can also use this guide to tailor Instance Monitor for your organization.

This chapter contains the following topics:

How This Guide is Organized

This guide contains the following information:

Chapter 1 Introduction	Contains information about: • Other documents available for Instance Monitor. • Contact Quest Software, Inc.
Chapter 2 Instance Monitor Concepts	Describes the basic features of the main Instance Monitor window.
Chapter 3 Getting Started With Instance Monitor	Contains tasks that affect how Instance Monitor displays and collects information.
Chapter 4 Desktop Features	Describes the various Instance Monitor features in detail.
Chapter 5 Configuring Instance Monitor	Contains tasks that you can use to tailor how Instance Monitor responds to the information it is collecting.
Chapter 6 Using Drilldowns	Drilldowns contain detailed information about your database. This chapter describes the drilldowns that are available and the information that is contained on them.
Chapter 7 Responding to Instance Monitor Alarms	Contains information about the alarms that are displayed in the Alarm Log drilldown.
Glossary	Contains Oracle and Instance Monitor terms and abbreviations.

Additional Guides and Resources

This guide contains information about installing and configuring Instance Monitor. It is one in a suite of documents provided with Instance Monitor. The following documents are included in the Instance Monitor documentation suite

Guide	Description
Instance Monitor Installation Guide	Contains information about installing Instance Monitor on your system.
Instance Monitor User's Guide	Provides general information about using Instance Monitor.
Instance Monitor Tuning Guide	Contains detailed information on how to use Instance Monitor to tune your Oracle database.

The documents are provided in hardcopy and also in Adobe® Acrobat® portable document format (PDF). The PDF files are included on the Instance Monitor installation CD. The CD also includes a copy of the Adobe® Acrobat® Reader. You must install the Reader before you can view the PDF files.

Using Help

Instance Monitor includes extensive help. You can activate this help by pressing the **F1** key on any screen in Instance Monitor. You can also view the help by selecting the Help menu and choosing **User Manual**.

The help window contains a number of navigation aids to assist you to find information. The following table shows the available navigation aids:

To display this....	Do this...
The table of contents.	Click the Contents tab.
The contents of a help topic.	Click the name of the topic. The contents of the topic display in the right pane.
All topics containing a particular word.	Follow these steps: **1** Click the Find tab. **2** Type the word you want to search for and press ENTER. The topics that contain the word are shown in the left pane. **3** Click the name of the topic to display the contents of the topic in the right pane.
All topics containing a phrase.	Follow these steps: **1** Click the Find tab. **2** Type the first word in the phrase. Select **And** or **Or**. **3** Type the second word in the phrase. **4** Repeat steps 2 and 3 until you have typed the complete phrase and press ENTER. The topics that contain the phrase are shown in the left pane. **5** Click the name of the topic to display the contents of the topic in the right pane.
Index entries.	Click the Index tab. A list of index entries are shown. Type the first letter of the term that you want to display. The list of index entries moves up as you type.
Glossary entries	Click the Glossary button at the top of the right pane. A list of terms in the glossary appears in the left pane. You can scroll through the list of terms. Click on the term to display the definition.

About Quest Software

Quest Software, Inc. is a leading provider of database management, output management, and high-availability software solutions for distributed computing environments. Quest Software's comprehensive and innovative software solutions allow organizations to achieve increased productivity and efficiency with immediate results. With headquarters in Newport Beach, California, Quest Software also has offices across the U.S. and in Germany, France, the U.K., and Australia.

For more information, contact us at:

610 Newport Center Drive, Suite 1400
Newport Beach, CA 92660-6565

Contacting Quest Software

Feel free to contact Quest Software for product information and class schedules. You can reach our company headquarters in the United States in any of the following ways:

Telephone:	(949) 720–1434
Fax:	(949) 720–0426
E-mail Sales & Marketing:	info@quests.com
Home page:	http://www.quest.com

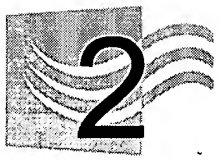
For information on our international offices, contact us at company headquarters.

If you have questions about using a Quest product, call our technical support staff Monday through Friday from 8:00 A.M. to 5:00 P.M. (PST). Please have the version number handy. If your question is about an error message, write the message down and have it available for the technical support representative.

Whenever you have a question, click **Support Bundle** on the Help menu. This creates a file called **support.zip** in your Instance Monitor directory. This file contains a snapshot of your Instance Monitor installation at the time the error occurred. E-mail this file to Quest Software with any request for assistance.

Telephone:	(949) 720–1434
Fax:	(949) 720–0426
E-mail Sales & Marketing:	support@quests.com

We appreciate your comments and suggestions regarding the printed and online documentation provided with this product. Please e-mail your comments and suggestions to documentation@quests.com.



Instance Monitor Concepts

The main Instance Monitor window contains a lot of information about what is happening in your database. This chapter describes the information in each section of the main window.

Drilldowns are also available from the main window. Drilldowns contain detailed information about top SQL, top sessions, database activity, database I/O, and alarms.

For more information see:

Understanding the main window

When you connect to a database, the main Instance Monitor window shows an overview of the database server. The window shows statistics for:

- Database processes
- Database-specific memory usage
- Database size
- Dataflows into and out of the database
- The number of connected users.

Instance Monitor updates the statistics and flows in real-time. Related database statistics are grouped together on panels. The groupings reflect how your Oracle database works. For more information about panels see *Understanding panels* on page 12.

Icons

The icons in Instance Monitor fall into the following categories:

Process icons are oval in shape and contain a single value that represents the state or existence of a database process.

Memory icons are rectangular and show the utilization of database-specific areas in memory.

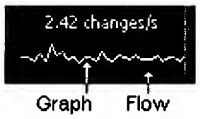
Disk icons are cylindrical and fill up as files increase in size.

Meters show a measurement. The highest and lowest possible values of the measurement are shown.

Dataflows

Each panel of icons is connected to other panels by a series of dataflows. These lines illustrate the rate (in seconds) at which data is moving through the system.

You can display a dataflow as a pulse or as a flow and a graph.



The flow shows you the current level of activity. As the rate of data transfer increases, so too does the speed of the flow. If the statistic represented by the flow moves into another threshold, the flow may change color. The combination of movement and color makes it easy to spot congested areas.

The graph sits on top of the flow and shows you how the load has varied over time.

The pulse moves in the direction of the dataflow. As the rate of data transfer increases, so too does the speed of the pulse.

The pulse can change color if the statistic represented by the pulse moves into another threshold.

Labels

Labels are shown above most icons and dataflows.

A label may have different metrics and thresholds to the component it is over. You can also tailor the metrics and thresholds of the labels.

Understanding metrics, thresholds, and severities

Before you start using Instance Monitor it is important to understand how metrics, thresholds, and severities are interrelated. You should also understand the relationship between these concepts.

What is a metric?

A metric is a column from a query that is run against a database. The query runs each time the Instance Monitor window is refreshed.

The query provides a performance statistic about the database (for example, the number of active users on the system, as a percentage of all users on the system). The query returns a value to Instance Monitor.

What is a threshold?

Each metric is divided into thresholds. Some metrics may have only one threshold, while others have many. A threshold is simply a range of values that can be returned by the metric. A threshold cannot belong to more than one metric.

The start range is the only value that is defined for a threshold. The end of the range is the value of the next threshold. If the threshold is the last one for the metric it does not have an end value.

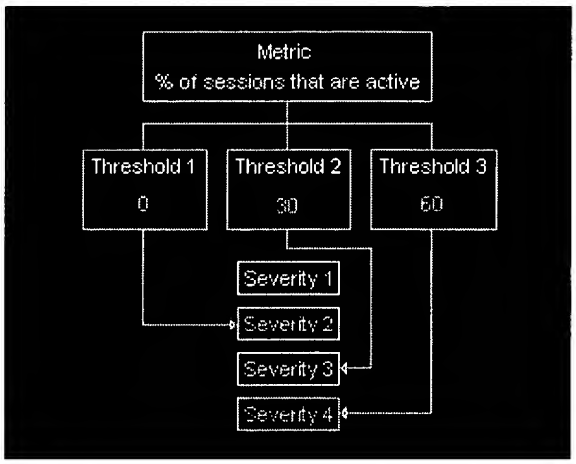
What is a severity?

Each threshold uses a severity. The severity controls what action Instance Monitor is to take when the value returned by a metric falls into the range controlled by a threshold. For example, you may set the severity to display a color, flash the color, emit a sound, or perform an action.

Severities may be attached to more than one threshold in the same metric. Severities can also be attached to thresholds in other metrics.

How do they work together?

The following diagram provides an example of the relationship between metrics, thresholds, and severities.



In this example, the query **Percent of sessions that are active** is run against the database. The value can fall into one of three ranges (called thresholds). If the query returned the value 20%, the value would fall into the range covered by Threshold 1.

Threshold 1 is attached to Severity 2. In this example, Severity 2 is set to display in green. The component that the metric belongs to changes to green.

A severity does not need to be attached to any thresholds.

Understanding panels

The main Instance Monitor window provides a quick and intuitive representation of the rate of processing in the Oracle instance. It also highlights any obvious bottlenecks or problem areas.

Related information is grouped together in panels. The following topics provide information about each of these panels and the information that is shown on them:

- SQL*NET panel (see page 12)
- Server Processes panel (see page 13)
- SGA panel (see page 15)
- Background Processes panel (see page 16)
- Disk Devices panel (see page 18).

SQL*NET panel

The SQL*Net panel shows:

1 The total number of users connected to the database.

2 The number of users who are currently active.

3 The rate at which data is being transferred from the server across the SQL*NET interface.

4 The rate at which data is being transferred to the server across the SQL*NET interface.

5 Active users as a percentage of all users.

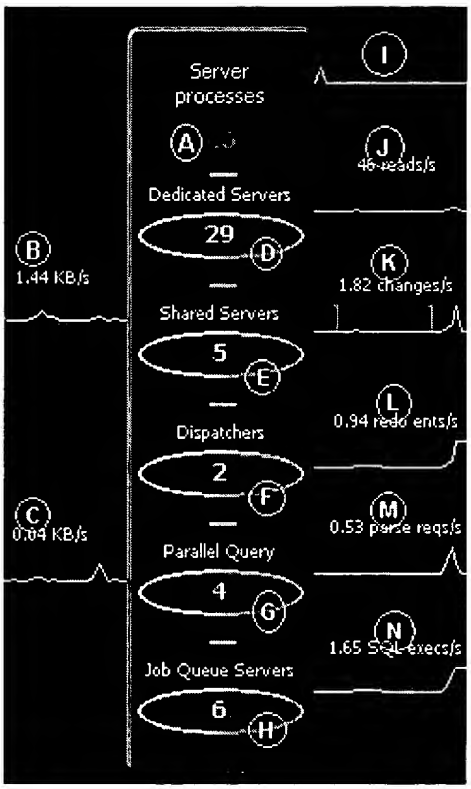
An alarm may be raised (**Active session** alarm) if the percentage of active sessions becomes too high.

Server Processes panel

The Server Processes panel shows the status of Oracle server processes. These processes are:

- Performing database activities on behalf of end users
- Mediating database connections.

The following diagram shows an example of the Server Processes panel. The circled letters on the panel identify the different parts of the panel. See the legend following the diagram for a description of the parts of the panel.



Legend

The Server Process panel is made up of the following parts:

A The number of server processes that are currently active.

B The rate at which data from the server processes is transmitted to the user processes.

C The rate at which data from user processes are transmitted to server processes.

D Dedicated server processes. Dedicated server processes perform work on behalf of a single client process. The number of servers varies as users log in and out of the database.

E Shared or multi-threaded servers (MTS). These perform work on behalf of more than one client process. The number of shared servers varies depending on load between the values of the configuration parameters MTS_SERVERS and MTS_MAX_SERVERS. If a high proportion of MTS are busy, then the **Multi-threaded server** alarm becomes active. See *Dealing with MTS contention* in the Instance Monitor Tuning Guide for advice on dealing with this alarm.

F MTS dispatchers. These coordinate the allocation of shared servers to client tasks. The number of dispatchers varies depending on the load between the values of the configuration parameters MTS_DISPATCHERS and MTS_MAX_DISPATCHERS. If a high proportion of MTS dispatchers become busy then an alarm may become current on this component. See *Dealing with MTS contention* in the Instance Monitor Tuning Guide for advice on dealing with this alarm.

G Parallel query servers support parallel execution of queries and (in Oracle8) DML statements. The number of servers varies depending on load between the configuration parameters PARALLEL_MIN_SERVERS and PARALLEL_MAX_SERVERS. If a high proportion of parallel servers become busy, then the **Parallel query server** alarm becomes current on this component.

H Job queue server processes. These are responsible for running PL/SQL commands submitted to the Oracle job queue via the DBMS_JOB package. The number of job queue processes is determined by the configuration parameter JOB_QUEUE_PROCESSES. If a high proportion of job queue servers become busy, then the **Parallel query server** alarm becomes current on this component.

I The rate at which blocks are read from disk by all server processes. The **Datafile read time** alarm becomes active if the average I/O time for these reads exceed a threshold.

J The rate at which blocks are read from the SGA by all server processes.

K The rate at which changes are made to blocks in the SGA by all server processes. The **Lock wait** alarm becomes current on this component if updates are being blocked by locks.

L The rate of redo buffer entries made by all server processes.

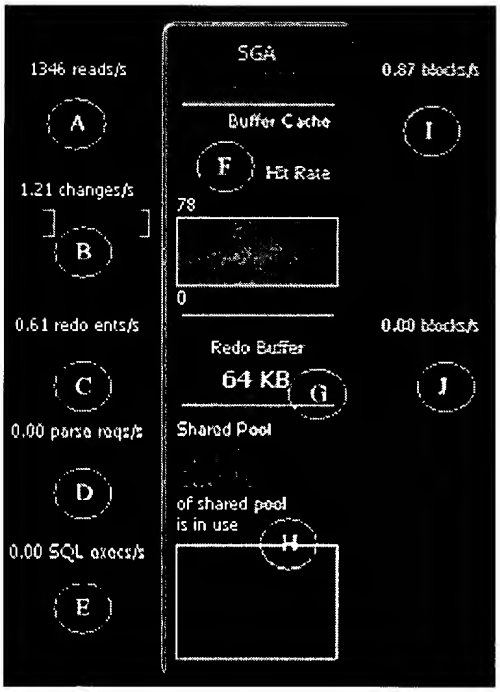
M The rate of SQL parse requests per second by all server processes. The **Parse ratio** alarm becomes current if the ratio of parse requests to execute requests exceeds a threshold.

N The rate of SQL execution requests per second by all server processes.

SGA panel

The SGA panel shows details of specific memory areas within the system global area (SGA). The SGA is an area of shared or common process memory that is used to cache frequently used data, SQL statements, procedures, and other structures.

The following diagram shows an example of the SGA panel. The circled letters on the panel identify the different parts of the panel. See the legend following the diagram for a description of the parts of the panel:



Legend

The SGA panel is made up of the following parts:

A The rate at which blocks are read from the SGA by all server processes.

B The rate at which changes are made to blocks in the SGA by all server processes. The **Lock wait** alarm becomes current on this component if updates are being blocked by locks.

C The rate of redo buffer entries made by all server processes.

D The rate of SQL parse requests per second by all server processes. The **Parse ratio** alarm becomes current if the ratio of parse requests to execute requests exceeds a threshold.

E The rate of SQL execution requests per second by all server processes.

F The buffer cache, which caches frequently accessed data blocks to avoid a disk I/O if these data blocks are subsequently required. The size of the buffer cache is controlled by the parameter DB_BLOCK_BUFFERS. Also shown is the buffer cache hit rate. A number of alarms may become current on this component. They include the **Buffer cache miss** alarm, **Cache buffer LRU chains latch** alarm, and the **Buffer busy wait** alarm.

G The redo buffer contains redo entries that must eventually be written to the redo log. Alarms can become current if processes spend time waiting for space in the redo buffer (**Log buffer space wait** alarm) or for redo buffer latches (**Redo allocation and copy latch** alarms).

H The shared pool caches SQL statements, PL/SQL programs, object definitions, and session memory for MTS sessions. Alarms can become current on this component if a low hit rate is observed for cached SQL statements (**SQL cache miss rate** alarm).

I Shows the rate at which blocks are being written by the Database Writer (DBWR) process

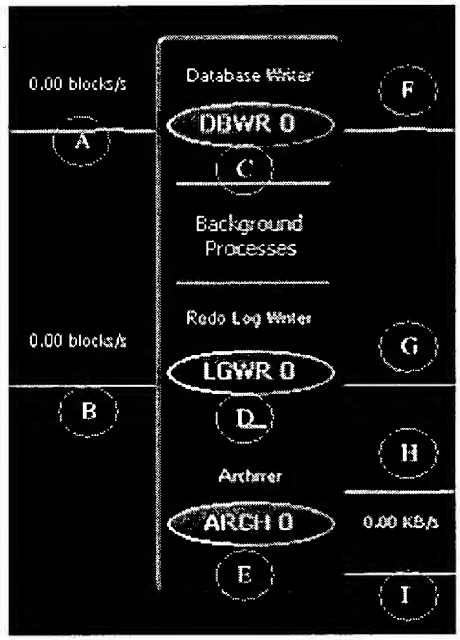
J Shows the rate at which redo log blocks are being written by the Redo Log Writer (LGWR) process.

Background Processes panel

The Background Processes panel displays the following key Oracle background processes:

- Database Writer (DBWR)
- Redo Log Writer (LGWR)
- Archiver (ARCH).

The following diagram shows an example of the Background Processes panel. The circled letters on the panel identify the different parts of the panel. See the legend following the diagram for a description of the parts of the panel:



Legend

The Background Processes panel is made up of the following parts:

A & F Rate at which modified blocks are written from the SGA to disk by the DBWR processes.

B & G Rate at which redo log entries are written to the redo log files by the LGWR processes.

C The DBWR process. The number of DBWR processes is determined by the parameter DB_WRITERS (Oracle7) or DB_WRITER_PROCESSES (Oracle8 and Oracle8i). This process may display an alarm if it is determined that the DBWR process is causing delays to other processes. The alarms are **Free buffer waits** alarm and **Write complete wait** alarm.

D The log writer process (LGWR). The number of LGWR processes is set at 1 in Oracle7. It may be configured in Oracle8 using the LGWR_IO_SLAVES parameter. An alarm can become current on this process if log switch waits occur (**Log switch time** alarm) or if the average redo I/O time exceeds a threshold (**Average redo write time** alarm).

E The Archiver process (ARCH). The number of ARCH processes is set at 1 in Oracle7. It may be configured in Oracle8 using the ARCH_IO_SLAVES parameter.

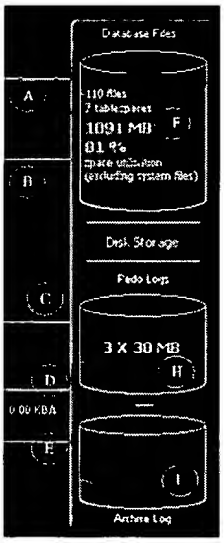
F See A.

G See B.

H & I Rate at which the archiver copies from redo log files to archived logs. This information is not available in Oracle7.

Disk Devices panel

The Disk Devices panel represents Oracle database and log files on disk.



The Disk Devices panel is made up of the following parts:

A The rate at which blocks are read from Oracle database files. The **Datafile read time** alarm activates if the average read time exceeds a threshold.

B The rate at which data blocks are written to the database files by the Database Writer (DBWR) process.

C The rate at which redo log entries are written to the redo logs by the redo log writer process (LGWR).

D & E The rate at which data is copied from redo logs into archive logs.

F Oracle database files. The level of fill in this component corresponds to the overall space utilization in the database. An alarm may become current (**Extent failure** alarm) if a segment is in imminent danger of failing to extend.

H Oracle redo logs. The fill level of the component corresponds to the number of archive logs that have not yet been archived (if archiving is enabled).

I The archive log destination.

Understanding drilldowns

The hierarchical design of Instance Monitor makes it possible for you to monitor your database at different levels of detail.

At its highest level (that is, the main Instance Monitor window) Instance Monitor displays a visual map of the major components in the current database. The detail at this level is designed to help you locate and identify bottlenecks quickly.

When you have isolated a problem, you can display a detailed breakdown of the underlying statistics. There are five types of information (called drilldowns). They are:

1 Top SQL

2 Top sessions

3 Activity

4 I/O

5 Alarm Log.

The are a number of ways to access the drilldowns. These include:

- Select the Navigator menu from the menu bar. Select the drilldown you want to display from this menu.
- Click on a component on the main Instance Monitor window. The most appropriate drilldown for that component appears.
- Click one of the following buttons to display a drilldown:

Displays the Top SQL drilldown.


Displays the Top Sessions drilldown.


Displays the Activity drilldown.

Displays the I/O drilldown.

Displays the Alarm Log drilldown.

For more information on drilldowns see *Using Drilldowns* on page 85.

Getting Started With Instance Monitor

This chapter contains information that you require to set-up Instance Monitor.

The chapter covers:

Setting-up users with the User Wizard

The Instance Monitor User Wizard is used to add users to the database that is to be monitored. It is also used to configure the accounts of existing database users so they can use Instance Monitor. The user must be configured for each database they are to monitor.

The User Wizard starts as soon as you run Instance Monitor for the first time. You can add more users to a database by choosing the Monitor menu and then selecting the **User Wizard** option. This task should be performed by the Database Administrator (DBA) of the database that is to be monitored. The DBA must log onto the database during the setup of users.

This task should be performed by the Database Administrator (DBA) of the database that is to be monitored. The DBA must log onto the database during the setup of users.

The screens that you see in the User Wizard depend on whether the user has been added to the database before. For more information see:

- *To create new users* on page 22. Use this method if the user does not have an account on the database.
- *To prepare existing users* on page 23. Use this method if the user has an existing account on the database but has not been configured to use Instance Monitor.
- *Upgrading Existing Users* on page 24. Use this method if you are upgrading your version of Instance Monitor. You can also use this method if a user has logged on to Instance Monitor but has not been configured to use Instance Monitor. (Users can only log on to Instance Monitor if they have an account in the database.)

To create new users

Follow these steps to create a new user in the database:

1 Click **Next** on the first user initialization window.

2 Type the SQL*Net connection string and the password for the SYS or DBA account. Click **Next**.

3 Select the **Create a new user** button and click **Next**.

4 Complete the following details for the new user and click **Next**.

User name	The user ID for the new user. You can type up to 30 characters.
Password	The password for the user. You can type up to 30 characters.
Confirm password	The password exactly as you typed it in the **Password** field.
Select any table	Select this check box if the user can generate explain plans for SQL statements created by other users.
Alter system	Select this check box if the user can issue trace commands or kill sessions.

5 A list of tablespaces that can be used for Instance Monitor tables and temporary segments appears. Select the tablespaces that are to be used and click **Next**.

6 Instance Monitor creates the user account. A meter displays the progress of the creation. Once the account is created click **Next**.

7 Once the account is configured, the User Wizard gives you the option of logging on to Instance Monitor using the new user account. Choose one of the following options:

- To log on as the new user select the **Log on as...** check box and click **Finish**.
- To log on as a different user (or perform another task) clear the **Log on as...** check box and click **Finish**. You can log onto Instance Monitor using another user ID by selecting the Monitor menu and clicking **Connect**.

To prepare existing users

Use this procedure if the user already has an account in the database that is to be monitored.

1 Click **Next** on the first user initialization window.

2 Type the SQL*Net connection string and the user name and password for the DBA account. Click **Next**.

3 Select the **Set up an existing user** button and click **Next**. A list of available users appears.

4 Highlight the user and type their password in the User password field. Click **Next**.

5 Instance Monitor prepares the user account. A meter displays the progress of the preparation. Once the account is created click **Next**.

6 Once the account is configured, the User Wizard gives you the option of logging on to Instance Monitor using the new user account. Choose one of the following options:

- To log on as the new user select the **Log on as...** check box and click **Finish**.
- To log on as a different user (or perform another task) clear the **Log on as...** check box and click **Finish**. You can log onto Instance Monitor using another user ID by selecting the Monitor menu and clicking **Connect**.

Upgrading Existing Users

This option is only available if Instance Monitor detects that the current user has been configured using an earlier version of Instance Monitor.

Use this procedure if:

- You are upgrading Instance Monitor to a new version.
- If a user logs on to Instance Monitor and that user has not previously been configured to work with Instance Monitor.

Follow these steps to upgrade the user:

1 Click **Next** on the first user initialization window.

2 Type the user name and password of the DBA account for the database and click **Next**.

3 Instance Monitor upgrades the user account. A meter displays the progress of the upgrade. Once the account is created click **Next**.

4 Once the account is configured, the User Wizard gives you the option of logging on to Instance Monitor using the new user account. Choose one of the following options:

- To log on as the new user select the **Log on as...** check box and click **Finish**.

- To log on as a different user (or perform another task) clear the **Log on as...** check box and click **Finish**. You can log onto Instance Monitor using another user ID by selecting the Monitor menu and clicking **Connect**.

Making a database available for monitoring

Use this procedure to establish a connection with a database. This task is included as part of the Instance Monitor User Wizard. If you used the Wizard and chose to connect to the database you can skip this procedure.

Ensure that the database is accessible to client applications. Use the appropriate connectivity software to establish a connection between your computer and the database server if a connection does not already exist.

The username and password that you use to connect to the database, must exist in the database. If you are unsure whether the user exists, contact your Database Administrator (DBA) or use the User Wizard to add a new user.

To add a database

1 On the Monitor menu, click **Connect**.

2 Complete the following fields on the **Connect** dialog box:

Login Username	Type your user ID.
Login Password	Type your password.
Connect String	The connect string used to link to a database (that is, the SQL*Net alias). The database name is defined within SQL*Net configuration utilities.

3 Choose one of the following options:

- Click **Cancel** to close the Connect dialog box. Any information you entered is discarded.
- Click **Connect** to connect Instance Monitor to the database. The standard queries are run on the database and the values are displayed on the main Instance Monitor window.

For more information about the objects in the main Instance Monitor window, see *Understanding the main window* on page 8.

Getting information about a panel or dataflow

To obtain more information about a component:

1 Position the cursor over the component. If more information is available, the mouse pointer changes to a hand. A label containing information about the component is displayed.

2 Click on the object to display the drilldown associated with the component.

If there is no information for the object, the mouse pointer does not change. Clicking on the object may display a drilldown.

Showing graphs as pulses

You can choose to display the flows on the main Instance Monitor screen as pulses or as graphs. When Instance Monitor first starts, the flows are displayed as graphs. Follow these steps to change the way flows are displayed:

1 On the Monitor menu, click **Options**.

2 Select the **Other** tab.

3 In the Flow settings section of the tab, choose one of the following options:

- To display flows as pulses, select the **Show flow graphs as pulses** check box.
- To display flows as graphs, clear the **Show flow graphs as pulses** check box.

4 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

This setting affects all flows on the main Instance Monitor screen.

Showing graph history

Most graphs on the Instance Monitor drilldowns show one or more statistics. They are plotted as values over time. You can specify how long Instance Monitor is to retain statistical information.

The graphs on the drilldowns progressively show the statistic over this period of time. When you start Instance Monitor for the first time, this option is set to one hour.

If you increase the graph history setting (for example, from 1 hour to 24 hours) only the data for the past hour is available initially. Instance Monitor progressively collects the information for the next 23 hours to display a full 24 hour period.

If you decrease the graph history setting (for example, from 24 hours to 1 hour) only the data for the past hour is retained. If you change the setting up again, Instance Monitor cannot access the historical information.

To change the graph history time

1 On the Monitor menu, click **Options**.

2 Select the **Drilldowns** tab.

3 In the **History settings** section of the tab, choose the graph history time from the drop-down list.

4 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

This setting affects all graphs that display a statistic over time.

Using fast initialize

The Fast Initialize dialog box appears when you connect to a database for the first time. If you select the **Do not show this message in future** check box, the Fast Initialize dialog box does not appear again. The setting you chose when you selected the check box is used every time you connect to a database.

If you choose to use fast initialize and the wait is for a file IO operation Instance Monitor does not display the name of the table or index to which the IO is occurring. This information is normally displayed on the **Session Details** and **Session Waits** tabs in the Top Sessions drilldown. See *Session Details tab* on page 87.

To change the fast initialize setting

1 On the Monitor menu, click **Options**.

2 Select the **Other** tab.

3 In the **Other settings** section of the tab, choose one of the following options:

- To use fast initialize mode every time you connect to a database, select the **Use "fast initialize" mode** check box.
- To use the normal (slow) initialize mode every time you connect to a database, clear the **Use "fast initialize" mode** check box.

4 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

Printing Instance Monitor drilldowns

You can print the graphs and other information shown in Instance Monitor.

To print a drilldown, follow these steps:

1 Display the drilldown you want to print (see *Understanding drilldowns* on page 19 for more information on selecting drilldowns).

2 On the Monitor menu, click **Print**. The standard Windows Print dialog box displays.

3 Select the printer where you want the report to print.

4 Click **OK**.

Starting Instance Monitor from a command line

Instance Monitor can be started from a command line. To start Instance Monitor, type the following command at the command line and press **Enter**:

```
IMONITOR.EXE
```

You can specify a number of switches that affect how Instance Monitor starts. The following switches are supported:

- `IMONITOR.EXE /connect:`*`USER/PASSWORD@CONNECT_STR`*

  Automatically connects to the specified database (connect_str) when Instance Monitor starts. For example:

  `IMONITOR.EXE /connect:iwmon/mypass@w804` logs the user `IWMON` onto the database `W804` using the password `MYPASS`.

- `IMONITOR.EXE /setup`

  Displays the user setup wizard rather than proceeding directly to the login screen. This setting overrides the `/CONNECT` command line switch.

- `IMONITOR.EXE /fast:n`

  `n` is a numeric switch. If `n` is not zero, starts Instance Monitor in fast initialize mode. If `n` is not specified the default value is 1 (start in fast initialize mode).

Both / and - are accepted in switches. For example, `/FAST` is equivalent to `-FAST`.

You can use more than one switch in a command. Separate the switches with a space. For example, `IMONITOR.EXE /SETUP /FAST:1`.

Exiting from Instance Monitor

On the Monitor menu, click **Exit** to close Instance Monitor. You can also click the standard Windows close control in the top right corner of the main Instance Monitor window. A warning message appears:

- To exit from Instance Monitor, click **Yes**.
- To cancel the exit and return to Instance Monitor, click **No**.

Instance Monitor prompts you to save the ***<connect_string>*.mpf** file if you have run a calibration, or made changes to the thresholds or severities.

Desktop Features

This chapter describes the major features of the Instance Monitor desktop. These include:

The main Instance Monitor window

The main Instance Monitor window shows a unique view of the activity of an Oracle instance. The display is based on the Oracle architecture diagram shown in *A review of the Oracle architecture* in the Instance Monitor Tuning Guide.

The Instance Monitor window helps you locate database bottlenecks quickly. Related database statistics are grouped together on panels that are connected by a series of graphical flows. Instance Monitor updates these flows in real-time so that you can see how quickly data is moving through the system.

The main elements of the Instance Monitor window are:

- The toolbar. Provides you with a quick way of navigating through different pages of information. For a detailed description of the options on the toolbar, see *Instance Monitor toolbar* on page 43.
- Panels. A group of related database statistics. The database statistics are shown as icons. The icons change color as their values move through the range of thresholds. For more information on panels see *Understanding panels* on page 12.
- Dataflows. A flow that illustrates the rate of data moving through the database. Dataflows change their speed and color, alerting you to performance issues. For more information see *Dataflows* on page 9.
- The connect string identifier. Identifies the database you are currently monitoring. Located in the top right corner of the window, the connect string identifier changes color to match the color of the component or dataflow that represents the worst severity. The following diagram shows an example of a connect string identifier:

Instance Monitor menus

You can use menus to access the functions of Instance Monitor. There are two types of menu in Instance Monitor. They are:

1 Standard menus. Accessed from the menu bar on the main Instance Monitor window. The available menus are:

- Monitor menu
- Navigator menu
- Tools menu
- Help menu.

2 Shortcut menus. These menus display when you click the right mouse button (right-click) on an object in the main Instance Monitor window.

Accessing standard menus

You can access a menu by clicking on it with the mouse. Most menus also have an access or accelerator key. To open a menu using the access key, press the ALT key and the underlined letter in the name of the menu, at the same time. The list of commands available from the menu appears.

For example, to display the Monitor menu you can click on Monitor in the menu bar or press ALT+M. The following diagram shows the menu bar with the Monitor menu selected:

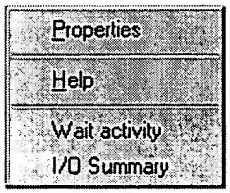
You can select a command from the menu by typing the letter that is underlined in the command name. For example, to display the Options window you could press the letter O when the Monitor menu is displayed. If the menu is not visible you can type the Monitor access key (ALT+M) and then press O.

Some commands also have a shortcut key. If you use the shortcut keys you do not have to display the menu first. Simply type the shortcut keys anywhere in Instance Monitor. Shortcut keys are shown on menus, after the name of the command (for example, F5 for **Refresh** on the Monitor menu).

Accessing shortcut menus

Shortcut menus contain a list of commands. The commands are relevant to the area of the screen where the cursor is positioned. For example, if you click the right mouse button over a dataflow, dataflow-specific commands display.

The following diagram shows an example of a shortcut menu:



The Threshold Wizard is accessed via a shortcut menu. On the shortcut menu, click **Properties**. The Threshold Wizard appears.

Monitor menu

The Monitor menu is accessed from the main menu bar. The Monitor menu contains the following commands:

Use this command...	To...
Connect	Establish a connection to a database.
Disconnect	Disconnects Instance Monitor from the current database.
Refresh	Update the data on the current screen.
Pause	Stops Instance Monitor sending queries to the database. Any queries in progress when you select this command are allowed to complete. If Instance Monitor is currently paused, this option changes to **Unpause**.
Save Page Settings	Saves the settings for the current page (***<connect_string>*.mpf** file).
Save All Settings	This option saves the ***<connect_string>*.mpf** file and the **monitor.ini** file.
Save As Default	Saves the ***<connect_string>*.mpf** file of the current database as **default.mpf**.
Edit Severities	Change the attributes of severities.
Options	Display the Pages and Other tabs on the Options window.
Calibration	Used to start, stop, cancel, or change a calibration.
User Wizard	Creates or configures a user's database account so they can use Instance Monitor.
Print	Prints the current drilldown. You cannot print the main screen.
Exit	Close Instance Monitor.

Navigator menu

The **Navigator** menu contains the following options:

Use this command...	To...
Top SQL	Display the Top SQL drilldown.
Top Sessions	Display the Top Sessions drilldown.
Activity	Display the Activity drilldown.
I/O	Display the I/O drilldown.
Alarm Log	Display the Alarm Log drilldown.
Home	Display the main Instance Monitor window.
Back	Move to the previous screen in the *Recently used list.*
Forward	Move to the next screen in the *Recently used list.*
Recently used list	Shows (up to) the last ten drilldowns you accessed. Click on the name of the drilldown you want to display.
Clear Navigator List	Select this option to clear the *Recently used list.*

For more information on drilldowns see *Understanding drilldowns* on page 19.

Tools menu

The Tools menu is accessed from the main menu bar. It contains the following options:

Use this command...	To...
SQLab Xpert	Start SQLab Xpert. SQLab Xpert provides context-sensitive tuning advice for SQL statements based on the Oracle execution plan and the database structure. Clicking on this option starts SQLab Xpert. For this link to work, SQLab Xpert 3.1c (or greater) must be installed on your PC and you must have DBA privileges for the database you are monitoring.

Use this command...	To...
Explain Plan	Display the Explain Plan. The Explain Plan allows you to determine the execution plan Oracle applies to a particular SQL statement. Instance Monitor allows you to view graphical representations of the execution plan for SQL statements being executed by a user or those identified by the Top SQL drilldown. Clicking on this option opens the Explain Plan. For more information on the Explain Plan see the Instance Monitor Tuning Guide.
Space Manager	Start Space Manager. Space Manager provides a comprehensive solution for space management and reorganization. Space Manager offers preventive maintenance, problem detection and resolution, and capacity planning across any number of databases. For this link to work, Space Manager 3.3 (or greater) must be installed on your PC and you must have DBA privileges for the database you are monitoring.

Help menu

The Help menu contains the following options:

Use this command...	To...
Context Help	Display context-sensitive help. The help relates to the information that is currently displayed on the screen.
User Manual	Display the table of contents for the online help system. This enables you to use the online help like an electronic version of the manuals.
Web Help	Link to Quest Software's Instance Monitor web page. From this page you can download upgrades or view frequently asked questions about Instance Monitor. You must have browser software installed and an active connection to the Internet.

Use this command...	To...
Support Bundle	Create a file called **support.zip** in your Instance Monitor directory. The file contains a snapshot of your Instance Monitor installation at the time you selected this option. You should always save the page before selecting this option. This ensures that the support bundle contains the latest settings. Database information is not included in the support bundle. E-mail this file to Quest Software with any request for assistance. See *Contacting Quest Software* on page 6 for more information on getting assistance with Instance Monitor.
About	Display: • Copyright and version information about Instance Monitor. • Your authorization details. You can also change your Authorization string from this window (for example, if you are upgrading from a trial version to a production version).

Instance Monitor toolbar

The Instance Monitor toolbar is located at the top of the main window. The toolbar provides you with quick access to commonly used commands and functions. Click once on a button on the toolbar to carry out the command.

To see a description of what each button does, rest your mouse pointer over the button. Information about the button appears in a Tooltip beneath the button.

If a button is gray, the command is currently unavailable.

The following table shows the buttons and what they are used for:

Go back to the previous screen or drilldown. You can select the Navigator menu to see a list of screens that have been displayed.

Go to the next screen or drilldown. This option is only available if you have gone back to a previous screen or drilldown.

Pause the current display. Instance Monitor does not submit queries to the database. Details on the screens or drilldowns are not updated until you click the **Pause** button again.

(or select **Unpause** from the Monitor menu).

Display the main Instance Monitor window for the current database.

Refresh the current screen (equivalent to selecting the **Refresh** option on the Monitor menu).

Displays the Top Sessions drilldown. This is equivalent to selecting the **Top Sessions** option on the Navigator menu.

Displays the Top SQL drilldown. This is equivalent to selecting the **Top SQL** option on the Navigator menu.

Displays the Activity drilldown. This is equivalent to selecting the **Activity** option on the Navigator menu.

Displays the I/O drilldown This is equivalent to selecting the **I/O** option on the Navigator menu.

Displays the Alarm Log. This is equivalent to selecting the **Alarm Log** option on the Navigator menu.

Displays context sensitive on-line help for Instance Monitor. This is equivalent to pressing **F1**.

Pausing and unpausing Instance Monitor

You can pause Instance Monitor so that the details you are viewing are not refreshed.


Pause button.

To pause Instance Monitor

Click the **Pause** button. Any queries in progress when you click this button are allowed to complete. No further queries are submitted to the database until you click this button again.

Note that pausing may affect:

- Calibration. If a calibration is in progress when the Pause is started, it continues to run. The calibration finishes at the time it was originally set to finish (it is not delayed by the Pause).
- Drilldowns. Any drilldown that gathers information when it is started (such as Top SQL) is affected by Pause. If you are at the main Instance Monitor window when you start Pause, and then you start the Top SQL drilldown, no data is displayed.


The flows on the main Instance Monitor window still move when pause is on. This indicates the speed of the flows at the time Pause was clicked.

To restart Instance Monitor

Click the **Pause** button or click **Unpause** on the Monitor menu. Queries are resubmitted and the main window and drilldowns update as usual. Events that occurred during the period Instance Monitor was paused are not displayed.

Metric Editor window

The Metric Editor window contains the metrics and thresholds for a component. You can use this window to set and modify the metrics and thresholds.

To access the Metric Editor window, follow these steps:

1 Position the cursor over a component.

2 Click the right mouse-button.

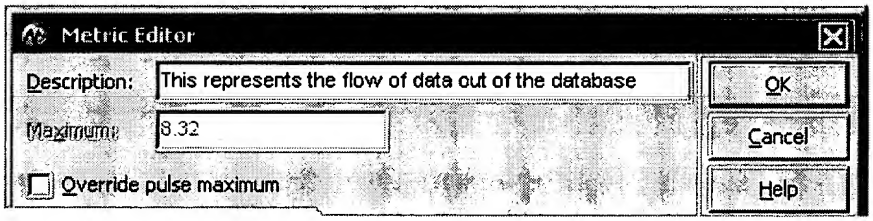
3 On the shortcut menu, click **Properties**.

4 On the Threshold Wizard click the **Advanced** button. The Metric Editor window appears.

The metrics and thresholds shown in the Metric Editor window may vary from one panel to another and from one dataflow to another. The metrics and thresholds are unique to a component. For this reason, the cursor must be positioned over a component before you can display the Metric Editor window.

The Metric Editor window contains three separate pieces of information. They are:

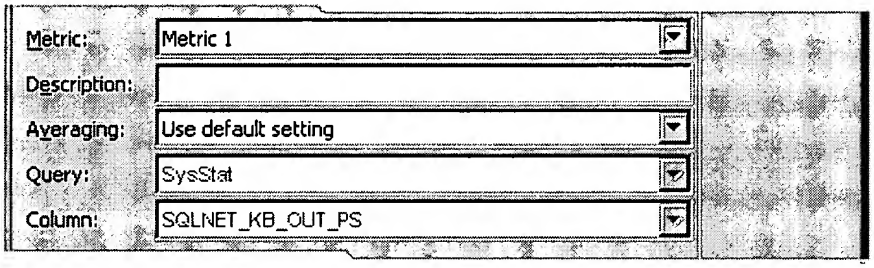
1 Component details. This is the top section of the Metric Editor window and contains the following fields:



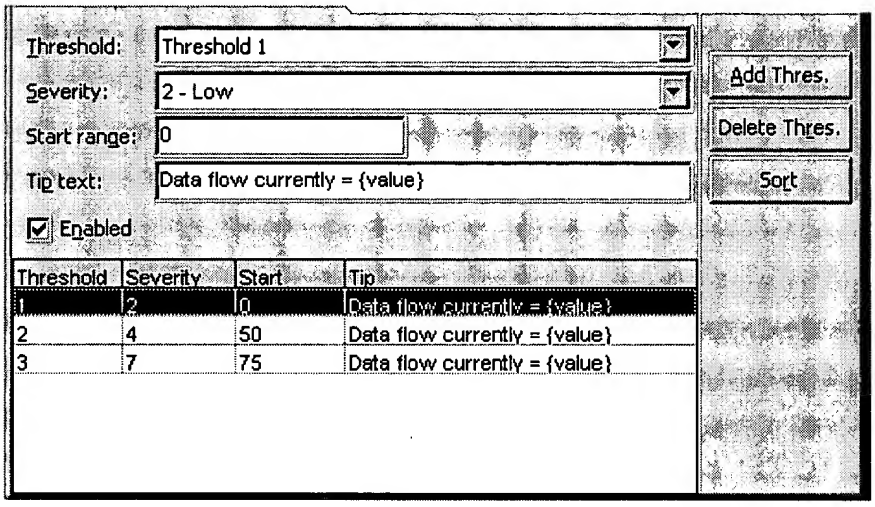
For more information on changing metric settings see:

- To change the description of a component on page 73
- To change the maximum value for a metric on page 74.

2 Metric details. The next five fields (from **Metric** to **Column**) show the settings for the metric that is currently active. You can choose another metric by clicking on the drop-down list in the **Metric** field and selecting it from the list. For more information on changing metric settings see *Managing metrics* on page 73.



3 Threshold details. The remaining fields show the thresholds that are attached to this metric. You can see the specific details of a threshold by clicking on it in the list or by clicking on the drop-down list in the **Threshold** field and selecting it from the list. For more information on changing threshold settings from the Metric Editor window see Changing the thresholds for a metric on page 76.



Options Window

The Options window is used to set most of the options that affect the Instance Monitor display.

On the Monitor menu, click **Options** to display the Options window. The window contains the following tabs:

- **Alarm** tab. Used to set alarm log options, including the location of the alarm log file.
- **Calibration** tab. Used to start and stop a calibration, and to set or change the length of time calibration is to run.
- **Drilldowns** tab. Used to set how long graph histories are to be retained and how often the Top Sessions drilldown is to be reset.
- **Other** tab. Used to set background and foreground refresh rates, to turn sounds on and off, and to set actions.

To select a tab, click on the name of the tab in the Options window.

Alarm tab

You can use the **Alarm** tab on the Options window to set what actions are to occur when an alarm is logged. Any alarms that have occurred during the current session are displayed on the Alarm Log window.

To change the alarm options

1 Complete the following fields on the **Alarms** tab:

Log alarms for events of the following severity or greater:	Select the lowest severity level that is to be reported to the alarm log. A metric that returns a value in a threshold that is set to use this severity (or a higher severity) is reported to the alarm log. The lowest severity that can be reported is Severity 3. The highest is Severity 7 or any higher severities you may define.
Log alarms to the Alarm Log window:	Select this check box if alarms are to be displayed in the Alarm Log window. If you do not select this check box the Alarm Log window is not updated.

Log alarms to the following file:	Select this check box if you want to log alarms to a file. You must type the filename (including path) in the text box below this field. You can select the path and file by clicking the browse button.
When Instance Monitor starts:	Choose one of the following options: • **Append new alarms to the end of the log file** Continues adding log entries to the end of the file specified in the **Log alarms to the following file** field. The Alarm log file can be viewed using an editor such as Notepad. • **Erase any previous log entries** Creates a new log file each time Instance Monitor starts. The Alarm Log window only displays alarms that have occurred since Instance Monitor was started.

2 Choose one of the following options:

- To save your changes, click **OK**. The new values for the alarm log are used the next time Instance Monitor needs to log an alarm.
- To close the window without saving your changes, click **Cancel**.

Calibration tab

Calibration resets the maximum value of the pulses so that they move properly in relation to the data Instance Monitor collects for a particular database.

Use the **Calibration** tab on the Options window to start, stop, change, or cancel a calibration.

For more information see:

- *To start a calibration* on page 58
- *To stop a calibration* on page 59
- *To change a calibration while in progress* on page 59.

Drilldowns tab

Use the **Drilldowns** tab to set:

- The length of time graph histories are to be retained.
- The reset options for the Top Sessions drilldown.

To change the History settings

1 In the **History settings** section of the **Drilldowns** tab, choose the graph history time from the drop-down list.

2 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

This setting affects all graphs that display a statistic over time. For more information see *Showing graph history* on page 29.

To change the Top Sessions reset interval

To improve performance, Instance Monitor only retrieves changed session details in the Top Sessions drilldown. Disconnected sessions may continue to be displayed until a cleanup occurs. A cleanup occurs every five minutes by default.

Cleaning up more frequently reduces the chance that disconnected sessions will be displayed. However, it may result in reduced performance and increased system overhead, especially on systems with many hundreds of users.

The options in this section of the **Drilldowns** tab are used to specify how often the Top Sessions window is fully refreshed.

Choose one of the following options:

- **Always reset Top Sessions.**

  Performs a full refresh of the Top Sessions window each time the window is refreshed. The refresh rate is set in the **Foreground refresh interval** field on the **Other** tab.

- **Reset Top Sessions every** *xxx* **minutes.**

  Performs a full refresh of the Top Sessions window at the interval you type (or choose) in the selection box. You can type or select values from 1 to 999 minutes.

Other tab

The **Other** tab on the Options window is used to set various options that affect the way Instance Monitor displays information. The tab is broken into the following sections:

Refresh settings

This section of the tab contains the settings for the background and foreground refresh rate. For more information see *To set the background or foreground refresh rate* on page 66.

Flow settings

This section contains settings that are used to:

- Set graph convergence. This option dampens major variations in a dataflow. See *To turn graph convergence* on page 65.
- Set whether graphs are to be displayed as a pulse or as a dataflow. For more information see *Showing graphs as pulses* on page 28.

Other settings

This section contains the following settings:

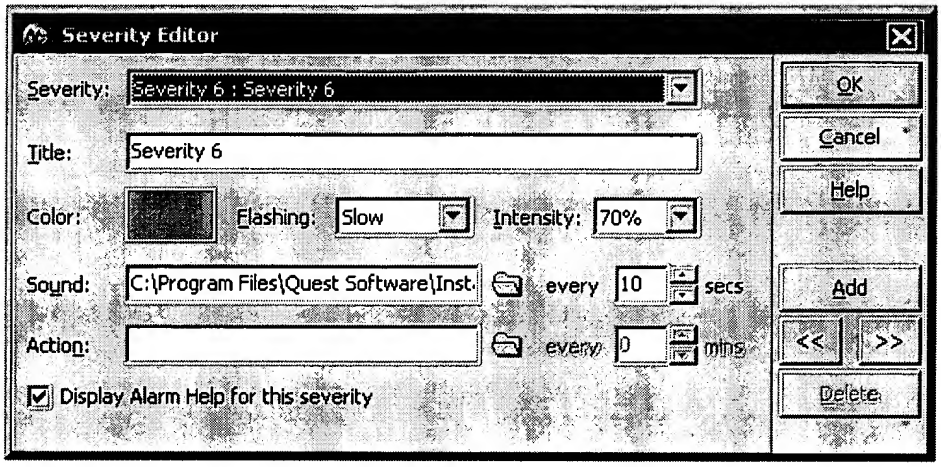
- **Enable alarm sounds**. If this option is selected, any audible sounds that are included in a severity can be heard. See *Turning sounds on or off* on page 67.
- **Use "fast initialize" mode**. If you choose to use fast initialize and the wait is for a file IO operation Instance Monitor does not display the name of the table or index to which the IO is occurring. This information is normally displayed on the **Session Details** and **Session Waits** tabs in the Top Sessions drilldown.

  For more information see the *Session Details tab* on page 87.

Severity Editor window

The Severity Editor window is used to add, delete, or modify the visual, audible, and other warnings that are used by Instance Monitor when a database metric reaches a severity level.

The following diagram shows an example of the Severity Editor window:



On the Monitor menu, click **Edit Severities**. The Severity Editor window appears.

For more information about severity levels see:

- *Managing severities* on page 80
- *Managing thresholds* on page 69.

Shortcut keys

Instance Monitor contains a number of shortcut keys. You can use these keys to run commands that are normally accessed from menus. In most cases the shortcut key can be used on any Instance Monitor screen.

The following table shows the Instance Monitor shortcut keys:

Use this key...	To do this...
ALT+Home	Display the main Instance Monitor window.
CTRL+← (left arrow)	Move to the previous screen in the Navigator list. For a list of screens in the list click the Navigator menu.
CTRL+→ (right arrow)	Move to the next screen in the Navigator list.
CTRL+A	Display the Activity drilldown.
CTRL+I	Display the I/O drilldown.
CTRL+L	Display the Alarm Log drilldown.
CTRL+Q	Display the Top SQL drilldown.
CTRL+S	Display the Top Sessions drilldown.
F1	Display context sensitive on-line help relating to the screen or drilldown you are currently viewing.
F5	Refresh the current screen.
ALT+F4	Exit from Instance Monitor.

Configuring Instance Monitor

This chapter describes the tools that you can use to configure Instance Monitor. These tools make it possible for you to tailor various options for the database you are monitoring.

If you run a calibration you are prompted to save the ***<connect_string>*.mpf** file. However, if you change other settings the new setting is not saved automatically. You must save the page or save all settings. See *Saving configuration details* on page 56.

This chapter contains the following information:

Saving configuration details

Whenever you change a configuration setting you should save the page. Instance Monitor provides three save options:

1 Save page settings.

2 Save all settings.

3 Save as default.

Save Page Settings

On the Monitor menu, click **Save Page Settings**.

This option saves the ***<connect_string>*.mpf** file for the page. This file contains:

- The maximum values of every dataflow graph on the page.
- The metrics and related thresholds for every object on the page.

The minimum value is stored but not used by Instance Monitor. Zero is used as the minimum value for every dataflow.

Save All Settings

On the Monitor menu, click **Save All Settings**. This saves the ***<connect_string>*.mpf** file and the **monitor.ini** file. The ***<connect_string>*.mpf** file contains:

- The maximum values of every dataflow graph on the page.
- The metrics and related thresholds for every object on the page.

The minimum value is stored but not used by Instance Monitor. Zero is used as the minimum value for every dataflow.

The **monitor.ini** contains:

- The connection string for the database, past connection strings, and the last user ID that was used to access the database. Passwords are not saved.
- Severity settings.
- The values of every property that is set on the **Options** tab.

- Table customizations (such as column names) for the Top Sessions drilldown.

Save as Default

On the Monitor menu, click **Save As Default** to save the current page as the **default.mpf** file.

This option saves the page settings (***<connect_string>*.mpf** file) as the default page (**default.mpf**). If you run Instance Monitor again (to monitor another database) the settings of this page are used.

This option performs the following steps:

1 Saves the page (this is the same as doing a **Save Page Settings**).

2 Saves the page as default.mpf.

If you have already monitored a database, a ***<connect_string>*.mpf** file exists. The settings of the **default.mpf** file are not used. If you want to use the settings of the **default.mpf** file, delete the ***<connect_string>*.mpf** file for the database. Instance Monitor uses the **default.mpf** the next time you connect to this database.

Calibrating Instance Monitor

The dataflows in Instance Monitor show the rate of data transfer over time. The speed of the dataflow varies based on the volume of data being transferred. The greater the traffic, the faster the dataflow.

For the dataflows to be an accurate representation of database activity, it is important that Instance Monitor knows the normal range of values for your database. For example, if the normal range of values for your database is from 0 to 100, a value of 8 is slow. However, if the range of values for your database is from 0 to 10, a value of 8 is very fast and the dataflow moves much faster.

Instance Monitor comes with a calibration tool that calculates the normal range of values for your database. You must place Instance Monitor in calibration mode for a set period of time. While in calibration mode, Instance Monitor measures the flow of data throughout the database. It uses this data to set the upper limit of each dataflow. You can accept the results or adjust them if necessary.

Instance Monitor always uses zero as the lowest value when displaying dataflows.

There are two ways of calibrating your database. They are:

1. By using the highest value.
2. By collecting the highest value and calculating the standard deviation. This method assumes a normal distribution of the data and smoothes out any spikes or anomalies.

When the calibration is complete, you are prompted to save the page. This ensures that the values are retained even if your computer fails. These values are used until you alter them or until you perform another calibration.

While the calibration is in progress, the word **Calibrating** appears at the bottom of the main Instance Monitor window.

To start a calibration

1. On the Monitor menu, click **Calibration** and then select **Start**. The Options window is displayed with the **Calibration** tab selected.
2. Select the **Calibration Mode** check box.

3 Use the scroll arrows in the **run for x hours** field to set the period of time calibration is to run. The minimum length of time calibration must run is one hour. You should run calibration during a period of typical processing for your database. Calibration turns off after the period of time specified in this field. You are prompted to save the page when calibration ends.

4 Select one of the following types of calibration:

- Select the **Straight** checkbox to calibrate using the highest value. Instance Monitor uses the highest value it encounters during calibration to set the highest point of activity for your database. Zero is always used as the lowest point of activity.
- Select the **Standard Deviation** checkbox to collect the highest value and calculate the standard deviation. This method assumes a normal distribution of the data and smoothes out any spikes or anomalies.

5 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

If your system crashes while calibration is in progress, any information that the calibration has gathered is lost. You must start calibration again.

To stop a calibration

1 On the Monitor menu, click **Calibration**.

2 Choose one of the following options:

- Click **Cancel** to stop the calibration immediately. Any values that have been gathered are discarded.
- Click **Stop** to end the calibration and save the values that have been gathered.

3 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

To change a calibration while in progress

1 On the Monitor menu, click **Calibration** and then click **Change**.

2 Use the scroll arrows in the **run for x hours** field to change the period of time calibration is to run. The value in this field represents the number of hours from the start of the calibration. For example, if you started calibration at 12 noon and set it to run for 4 hours it would finish at 4pm. If you change the value in this field to 3, calibration finishes at 3pm.

3 You can change the calibration type. Select one of the following types of calibration:

- Choose **Straight** to calibrate using the highest value. Instance Monitor uses the highest value it encounters during calibration to set the highest point of activity for your database. Zero is used as the lowest point of activity.
- Choose **Standard Deviation** to collect the highest value and calculate the standard deviation. This method assumes a normal distribution of the data and smoothes out any spikes or anomalies.

If you change the calibration type while calibration is in progress, Instance Monitor changes from when the change was made. The new method is not applied to the data that was collected previously.

4 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

To override the calibrated values for a dataflow

Use this procedure to override the maximum calibrated value for a dataflow. This procedure only affects the dataflow you select. You can only change the maximum value. The minimum value is always zero.

You cannot use this procedure while Instance Monitor is in calibration mode (that is, while the word **Calibrating** is shown at the bottom of the main Instance Monitor window).

Follow these steps to override the maximum values for a dataflow:

1 Right-click on a dataflow. On the shortcut menu, click **Properties**.

2 On the Threshold Wizard click the **Advanced** button. The Metric Editor window appears.

3 Select the **Override Pulse Maximum** check box.

4 In the **Maximum** field, type the upper limit of the dataflow's y-axis.

5 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

To cancel the results of a calibration

On the Monitor menu, click **Calibration**, then click **Cancel**. The calibration stops but the values are not saved. The values of the previous calibration are used. If no calibration has been done before, the default values are used. This can only be run while the calibration is in progress.

Coping with anomalies and spikes in a graph

One of the major features of Instance Monitor is its user interface. At a glance you can tell the status of your database. From time to time anomalies and spikes in dataflows can cause unusual graphs. Instance Monitor provides two features, Moving Averages and Graph Convergence, to help you smooth out anomalies and spikes.

Moving averages

You can define the number of points in a dataflow that are to be averaged. This smoothes the peaks and troughs in your graph and gives an overall trend of the dataflow.

You can set a default moving average for Instance Monitor. You can decide if a metric is to use the default moving average, a different moving average, or no moving average at all.

When you set the moving average, Instance Monitor adds a number of plot points together and divides the total by the moving average number. The number of points that are added together is the same as the moving average number. (A point is plotted each time the graph is refreshed.)

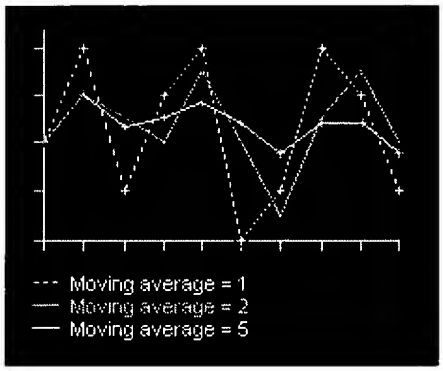
For example, the following table shows a dataflow. The actual dataflow would plot points at 2, 4, 1, and so on. This is the same as setting the moving average to 1.

Setting	Plot Points									
Actual dataflow	**2.0**	**4.0**	**1.0**	**3.0**	**4.0**	**0.0**	**1.0**	**4.0**	**3.0**	**1.0**
Moving average = 1	2.0	4.0	1.0	3.0	4.0	0.0	1.0	4.0	3.0	1.0
Moving average = 2	2.0	3.0	2.5	2.0	3.5	2.0	0.5	2.5	3.5	2.0
Moving average = 5	2.0	3.0	2.3	2.5	2.8	2.4	1.8	2.4	2.4	1.8

If the moving average was set to 2, Instance Monitor adds the last two plot points and divides by two. In the above example it would plot the third point at 2.5 ((4+1)/2). The oldest point is dropped from the calculation and the latest added.

If the moving average was set to 5, Instance Monitor would add the last five plot points and divide by five. In the above example it would plot the sixth point at 2.4 ((4+1+3+4+1)/5).

The following diagram shows how these flows would be plotted:



The higher the moving average, the less peaks and troughs are shown. Spikes and other anomalies are smoothed.

You must be careful not to set the moving average too high. If you set the moving average too high your graph may appear flat and may not be very meaningful. The maximum moving average in Instance Monitor is 10.

Where the moving average is greater than the number of points available for averaging, the points that are available are averaged using the following formula:

```
(Total of available points)/(Number of points)
```

For example, if the moving average is 5 and the first three values gathered are 2.0, 4.0, and 1.0, the following points are plotted:

- **2.0** (2/1)
- **3.0** ((2+4)/2)
- **2.3** ((2+4+1)/3)

To set the default moving average

1 On the Monitor menu, click **Options**.

2 Click the **Other** tab on the Options window.

3 In the **Metric Smoothing** section of the tab, select a moving average in the **Default setting** drop-down list.

4 Choose one of the following options:

- To save your changes, click **OK**. The moving average you entered is used to display your graphs from the next refresh.
- To close the window without saving your changes, click **Cancel**.

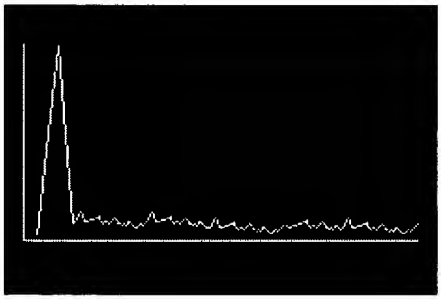
If you require more information about moving averages see *Coping with anomalies and spikes in a graph* on page 62.

If you make a change to the moving average settings, the change is effective from the next refresh. Existing graphs are not redrawn using the new settings.

Moving averages can be turned on or off for each metric. For more information see To set the default moving average on page 63.

Converging graphs

Graphs in Instance Monitor are normally focussed so they show the highest and lowest values that have been encountered in a dataflow. In the event of a spike or trough, the graph is resized to show the top of the spike or the bottom of the trough. If your normal dataflow is in the range 100 to 250 and the spike is at 1000, the normal dataflow is compressed in the graph so the spike can be displayed. The following figure shows an example of this type of dataflow.



The data on the graph after the spike is meaningless. In fact, what may be occurring is that the dataflow is varying widely in the normal range between 100 and 250.

If you choose to converge your graphs, the effect of spikes and troughs on the graph is reduced. The focus of the graph shrinks each time a new point is plotted.

Over time, the focus of the graph shrinks so that the normal range of the dataflow is visible and meaningful. The top of the spike or bottom of the trough is not shown on the graph.

The focus of the graph keeps shrinking until a new spike or trough exceeds the display. When this occurs, the graph is reset to display the new spike or trough and the convergence starts again.

To turn graph convergence on

1 On the Monitor menu, click **Options**.

2 Click the **Other** tab on the Options window.

3 Select the **Graph Converge** check box.

4 Select the percentage by which graphs are to be converged each time a point is plotted. You can enter any value between 0.00 and 100.00%.

5 Choose one of the following options:

- To save your changes, click **OK**. The graph convergence percentage you entered is used to converge your graphs from the next refresh.
- To close the window without saving your changes, click **Cancel**.

If you require more information about graph convergence see *Coping with anomalies and spikes in a graph* on page 62.

If you make a change to the graph convergence settings, the change is effective from the next refresh. Existing graphs are not redrawn using the new settings.

To turn graph convergence off

1 On the Monitor menu, click **Options**.

2 Click the **Other** tab on the Options window.

3 Clear the **Graph Converge** check box.

4 Choose one of the following options:

- To save your changes, click **OK**. Graph convergence is turned off from the next refresh.
- To close the window without saving your changes, click **Cancel**.

Changing the refresh interval

The information on Instance Monitor screens is refreshed regularly to give you a real-time view of the activity on your database. Screens you are viewing (foreground screens) are refreshed more frequently than screens you are not viewing (background screens). You can change the refresh rates for both types of screen.

To refresh the screen immediately

On the Monitor menu, click **Refresh**. You can also press **F5**. The current screen is updated immediately.

This option only affects the foreground screen. When you choose to refresh the screen, all queries are resubmitted.

To set the background or foreground refresh rate

1 On the Monitor menu, click **Options**.

2 Click the **Other** tab on the Options window.

3 Choose one of the following options:

- Select the foreground refresh rate in the **Foreground Refresh Interval** box. The minimum value is 1 second. The maximum value is 999 seconds.
- Select the background refresh rate in the **Background Refresh Interval** box. The minimum value is 10 seconds. The maximum value is 999 seconds.

4 Choose one of the following options:

- .To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

Turning sounds on or off

Use this procedure to turn sounds on or off.

If you use this option to turn sounds off, it overrides all other settings for sound. For example, if you set a sound for a severity it is not used unless this setting is on.

1 On the Monitor menu, click **Options**.

2 On the Options window click the **Other** tab.

3 Choose one of the following options:

- To turn sounds on, select the **Enable alarm sounds** check box.
- To turn sounds off, clear the **Enable alarm sounds** check box.

4 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

Removing a connection

You may decide that you no longer want to monitor a database. To remove a connection, follow these steps:

1 On the Monitor menu, click **Disconnect**. A warning message appears. Choose one of the following options:

- Click **Yes** to confirm the removal. The database connection is removed from the display and monitoring of the database ends.
- Click **No** to cancel the removal. The database connection is not removed and monitoring continues.

2 If you clicked **Yes** in the message you can:

- On the Monitor menu, click **Connect** to connect to another database. See *Making a database available for monitoring* on page 26.
- On the Monitor menu, click **Exit** to close Instance Monitor.

Managing thresholds

A threshold is a range of values that might be returned by a metric. A threshold indicates what severity is used when the metric is returning values in the range.

A threshold belongs to a metric. It cannot belong to more than one metric. However, a metric may have one or more thresholds.

Instance Monitor provides you with a Threshold Wizard that manages all of the threshold management tasks. You can also perform some of these tasks from the Metric Editor window.

To raise or lower a threshold in the order

1 Right-click on a component. On the shortcut menu, click **Properties**. The Threshold Wizard appears.

2 Select **Raise/Lower Current Threshold Limit** from the drop-down list and click **Next**.

3 Type or select the start range for this threshold.

4 You can also adjust the other thresholds that are attached to this metric by the same amount. The thresholds are adjusted by the value in the **Offset Value** field. To adjust the thresholds, select the checkbox and choose one of the following options:

- To adjust all thresholds below this threshold, select the **All thresholds with a start range less than this threshold** radio button.
- To adjust all thresholds above this threshold, select the **All thresholds with a start range greater than this threshold** radio button.
- To adjust all thresholds, select the **All thresholds** radio button.

5 Click **Next**.

6 Type a tip for the threshold and click **Finish**. You can enter up to 256 characters in this field.

To add a threshold

1 Right-click on a component. On the shortcut menu, click **Properties**. The Threshold Wizard appears.

2 Select **Add a Threshold** from the drop-down list and click **Next**.

3 Select the severity you want to use for this threshold from the drop-down list. The properties of the severity are shown at the bottom of the screen. Click **Next**.

4 Type a tip for the threshold and click **Next**. You can enter up to 256 characters in this field. You can also use the following variables in this field:

- **{value}** – Inserts the caption for the component.
- **{connect}** – Inserts the current connect string.
- **{range}** – Inserts the threshold start range.

The variable name must be typed in lower case.

For example, if a threshold starts at 50 and you type the following text in this field:

This threshold starts at {range}

The following text is displayed when the threshold is active:

This threshold starts at 50.

5 Type or select the start range for this threshold and click **Next**. If you enter a value that is already used by another threshold an error message appears.

6 If you do not want the metric to use this threshold, clear the **Enable this threshold** check box. The metric cannot use this threshold until it is enabled.

7 Click **Finish**. The main Instance Monitor window appears. If you enabled the threshold it is used the next time the metric returns a value in the range the threshold covers.

To delete a threshold

1 Right-click on a component. On the shortcut menu, click **Properties**. The Threshold Wizard appears.

2 Select **Delete a Threshold** from the drop-down list and click **Next**.

3 Select the threshold you want to delete and click **Finish**. The threshold is deleted immediately.

You cannot delete the last threshold in a metric.

To edit a threshold

1 Right-click on a component. On the shortcut menu, click **Threshold Adjustment**. The Threshold Wizard appears.

2 Select **Edit a Threshold** from the drop-down list and click **Next**.

3 Select the threshold you want to modify and click **Next**.

4 Select the severity you want to use for this threshold from the drop-down list. The properties of the severity are shown at the bottom of the screen. Click **Next**.

5 Type a tip for the threshold and click **Next**. You can enter up to 256 characters in this field.

6 Type or select the start range for this threshold and click **Next**. If you enter a value that is already used by another threshold an error message appears.

7 If you do not want the metric to use this threshold, clear the **Enable this threshold** check box. The metric cannot use this threshold until it is enabled.

8 Click **Finish**. The main Instance Monitor window appears. If you enabled the threshold it is used the next time the metric returns a value in the range the threshold covers.

To split a threshold

1 Right-click on a component. On the shortcut menu, click **Properties**. The Threshold Wizard appears.

2 Select **Split a Threshold** from the drop-down list and click **Next**.

3 Select the threshold you want to split and click **Next**.

4 Select the severity you want to use for this threshold from the drop-down list. The properties of the severity are shown at the bottom of the screen. Click **Next**.

5 Type a tip for the threshold and click **Next**. You can enter up to 256 characters in this field.

6 Type or select the start range for this threshold and click **Next**. If you enter a value that is already used by another threshold an error message appears.

7 If you do not want the metric to use this threshold, clear the **Enable this threshold** check box. The metric cannot use this threshold until it is enabled.

8 Click **Finish**. The main Instance Monitor window appears. If you enabled the threshold it is used the next time the metric returns a value in the range the threshold covers.

To enable or disable a threshold

1 At the main Instance Monitor window, right-click on a component.

2 On the shortcut menu, click **Properties**.

3 On the Threshold Wizard click the **Advanced** button. The Metric Editor window appears.

4 On the Metric Editor window, select the metric that contains the threshold you want to change.

5 Select the threshold you want to enable or disable from the drop-down list in the **Threshold** field.

6 Choose one of the following options:

- To enable the threshold, select the **Enabled** check box.
- To disable the threshold, clear the **Enabled** check box.

7 You can repeat this procedure to enable or disable other thresholds.

8 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

If a threshold is disabled it is ignored by Instance Monitor. If you disable a threshold, the values of the previous threshold are used. If there are no thresholds enabled, the settings of the lowest threshold are used.

Managing metrics

A metric is a column from a query that is run against a database. The query runs each time the Instance Monitor window is refreshed. The query provides a performance statistic about the database (for example, the number of active users on the system, as a percentage of all users on the system). The query returns a value to Instance Monitor.

Each metric is divided into thresholds. Some metrics may have only one threshold, while others have many. A threshold is simply a range of values that can be returned by the metric. A threshold cannot belong to more than one metric.

Each threshold uses a severity. The severity controls what action Instance Monitor is to take when the value returned by a metric falls into the range controlled by a threshold. For example, you may set the severity to display a color, flash the color, emit a sound, or perform an action.

To display a metric

1 Right-click on the component.

2 On the shortcut menu, click **Properties**.

3 On the Threshold Wizard click the **Advanced** button. The list of metrics and thresholds for this component appears in the Metric Editor window.

4 Use the drop-down list in the **Metric** field to select the metric you want to display or edit. The thresholds for this metric appear at the bottom of the Metric Editor window.

To change the description of a component

1 Right-click on a component.

2 On the shortcut menu, click **Properties**.

3 On the Threshold Wizard click the **Advanced** button.

4 Type a new description for the component in the first **Description** field on the Metric Editor window.

5 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

To change the maximum value for a metric

1. Right-click on a component.
2. On the shortcut menu, click **Properties**.
3. On the Threshold Wizard click the **Advanced** button.
4. On the Metric Editor window select the metric you want to change from the drop-down list in the **Metric** field.
5. Select the **Override pulse maximum** check box.
6. Type the new value in the Maximum field.
7. Choose one of the following options:
   - To save your changes, click **OK**.
   - To close the window without saving your changes, click **Cancel**.

If you change the maximum value, the new value overrides the calibrated value for this component. You cannot reverse this operation. This value is used as the maximum value until you change it again or until you run a calibration.

To change the description for a metric

1. Right-click on a component.
2. On the shortcut menu, click **Properties**.
3. On the Threshold Wizard click the **Advanced** button. The Metric Editor window appears. Select the metric you want to change from the drop-down list in the **Metric** field.
4. Type a new description for the metric in the second **Description** field on the Metric Editor window.
5. Choose one of the following options:
   - To save your changes, click **OK**.
   - To close the window without saving your changes, click **Cancel**.

Changing the moving average for a metric

You can set the moving average for an individual metric, use a default value, or turn moving averages off for the metric. For more information on moving averages see *Coping with anomalies and spikes in a graph* on page 62.

To use the default moving average for a metric

1 Right-click on a component and choose **Properties** from the shortcut menu.

2 On the Threshold Wizard click the **Advanced** button.

3 On the Metric Editor window select the metric you want to change from the drop-down list in the **Metric** field.

4 In the **Averaging** field, select **Use default setting**.

5 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

To turn moving averages off for a metric

1 Right-click on a component and choose **Properties** from the shortcut menu.

2 On the Threshold Wizard click the **Advanced** button. The Metric Editor window appears.

3 On the Metric Editor window select the metric you want to change from the drop-down list in the **Metric** field.

4 In the **Averaging** field, select **Don't average metric values**.

5 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

To set the moving average for an individual metric

1 Right-click on a component. On the shortcut menu, click **Properties**.

2 On the Threshold Wizard click the **Advanced** button. The Metric Editor window appears.

3 On the Metric Editor window select the metric you want to change from the drop-down list in the **Metric** field.

4 In the **Averaging** field, select one of the options beginning **Average over....** The number in the option indicates how many samples are to be used to calculate the moving average.

5 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

Changing the thresholds for a metric

Instance Monitor contains a Threshold Wizard which you can use to manage thresholds. Some threshold management tasks can be performed from the Metric Editor window. These include:

- Adding a threshold (see below)
- Deleting a threshold (see page 77)
- Enable or disable a threshold (see page 77)
- Change the details of a threshold (see page 78)
- Sorting a group of thresholds (see page 79).

To add a threshold using the Metric Editor

1 Right-click on a component. On the shortcut menu, click **Properties**.

2 On the Threshold Wizard click the **Advanced** button. The Metric Editor window appears.

3 On the Metric Editor window select the metric (from the drop-down list in the **Metric** field) to which you want to add a threshold.

4 Click the **Add Thres.** button. A new threshold is added to the end of the list of thresholds. It is given the next available threshold number. The default start value of the new threshold is one more than the start value of the last threshold.

5 You can:

- Select another severity for this threshold by using the drop-down list in the **Severity** field.

- Change the start range for this threshold by typing over the value in the **Start Range** field.
- Change the tip that displays when this threshold is active by typing text into the **Tip text** field. You can also use the following variables in this field:
  - **{value}** – Inserts the caption for the component.
  - **{connect}** – Inserts the current connect string.
  - **{range}** – Inserts the threshold start range.

  The variable name must be typed in lower case.
- Enable or disable the threshold by selecting or clearing the **Enabled** check box.

6 If the start ranges of thresholds are out of order, click the **Sort** button. The thresholds are reordered by start range value.

7 Choose one of the following options:
- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

To delete a threshold using the Metric Editor

1 Right-click on a component. On the shortcut menu, click **Properties**.

2 On the Threshold Wizard click the **Advanced** button. The Metric Editor window appears.

3 On the Metric Editor window select the metric (from the drop-down list in the **Metric** field) to which the threshold belongs.

4 Select the threshold you want to delete and click the **Delete Thres.** button. The threshold is removed.

5 Choose one of the following options:
- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

You cannot delete the last threshold in a metric.

To enable or disable a threshold using the Metric Editor

1. Right-click on a component. On the shortcut menu, click **Properties**.
2. On the Threshold Wizard click the **Advanced** button. The Metric Editor window appears.
3. On the Metric Editor window select the metric (from the drop-down list in the **Metric** field) to which the threshold belongs.
4. Select the threshold you want to enable or disable. Choose one of the following options:
   - To enable the threshold, select the **Enabled** check box.
   - To disable the threshold, clear the **Enabled** check box.
5. Choose one of the following options:
   - To save your changes, click **OK**.
   - To close the window without saving your changes, click **Cancel**.

To change a threshold using the Metric Editor

1. Right-click on a component. On the shortcut menu, click **Properties**.
2. On the Threshold Wizard click the **Advanced** button. The Metric Editor window appears.
3. On the Metric Editor window select the metric (from the drop-down list in the **Metric** field) that contains the threshold you want to change.
4. Select the threshold you want to change by selecting the threshold using the drop-down list in the **Threshold** field.
5. You can:
   - Select another severity for this threshold by using the drop-down list in the **Severity** field.
   - Change the start range for this threshold by typing over the value in the **Start Range** field.
   - Change the tip that displays when this threshold is active by typing text into the **Tip text** field. You can also use the following variables in this field:
     - **{value}** – Inserts the caption for the component.
     - **{connect}** – Inserts the current connect string.

- **{range}** – Inserts the threshold start range.

  The variable name must be typed in lower case.

- Enable or disable the threshold by selecting or clearing the **Enabled** check box.

6 If the start ranges of thresholds are out of order, click the **Sort** button. The thresholds are reordered by start range value.

7 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

To sort a group of thresholds

1 Right-click on a component. On the shortcut menu, click **Properties**.

2 On the Threshold Wizard click the **Advanced** button. The Metric Editor window appears.

3 On the Metric Editor window select the metric (from the drop-down list in the **Metric** field) that contains the thresholds you want to sort.

4 Click the **Sort** button. The thresholds are reordered by start range value.

5 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

Managing severities

Each threshold can be linked to a severity. The severity determines what action occurs when a threshold is reached.

The actions that can occur include:

- Changing the color of the component.
- Changing the speed and intensity at which the color flashes on and off (this option does not apply to dataflows).
- Emitting a sound.
- Performing an action such as sending an e-mail or pager message.

The properties of severities are managed from the Severity Editor window.

For information on linking a severity to a threshold see *Managing thresholds* on page 69.

To create a severity

1 On the Monitor menu, click **Edit Severities**. The Severity Editor window appears.

2 Click **Add**. The number of the next available severity appears in the **Severity** and **Title** fields.

3 Complete the following fields on the Severity Editor window:

Severity	This field defaults to the number of the next available severity.
Title	Type the title of the severity. The title is included as the last part of the Severity field (after the severity number).
Color	To change the color that appears when the severity is reached, click in the color box. The Color Selection window appears. Click on the color you want to display and then click **OK**. The Severity Editor window redisplays with the color you selected in the color box. You can also create your own colors if necessary.
Flashing	Select one of the following options from the drop-down list:

- **None**. The object does not flash if this severity is reached.
- **Slow**. The object flashes once every 0.6 seconds if this severity is reached.
- **Fast**. The object flashes once every 0.3 seconds if this severity is reached.

Intensity Select a number from the drop-down list. This field indicates the strength of the color when it is flashing. For example, a value of 10 indicates that the color should dim by 10% when it flashes. A value of 100 indicates that the color should dim by 100% (that is, it flashes between black and full strength).

Sound Type or browse for the location of a sound file (.wav) that is to run when the severity is reached.

Every Type how often (in seconds) the sound is to be played. If you do not enter a value in this field (that is, the value is zero), the sound is heard only once.

Action Type or select an action that is to be performed each time a severity is reached. You can use the following variables in this field:

- **{connect}** – Inserts the current connect string.
- **{name}** – Inserts the name of the component that is at this severity.

The variable name must be typed in lower case.

For example, if you wanted to open a specific notepad file when a severity is used you could enter the following text in the **Action** field:

notepad {name}.txt

If the SQL_Net component began to use this severity a notepad file called **sql_net.txt** would open.

Every Type how often (in minutes) the action is to be performed. If you do not enter a value in this field (that is, the value is zero), the action is performed once.

Display alarm help for this severity	If this check box is selected, the text in the metric description field is used instead of the component description. The component description is contained in the first **Description** field on the Metric Editor window (see page 46). The metric description is contained in the second **Description** field on the Metric Editor window.

4 Choose one of the following options:

- To save your changes, click **OK**. You can add the new severity to a threshold.
- To close the window without saving your changes, click **Cancel**.

To change the order of severities

1 On the Monitor menu, click **Edit Severities**. The Severity Editor window appears.

2 Use the drop-down list in the **Severity** field to locate the severity that you want to reposition.

3 Highlight the severity that you want to reposition and click the positioning buttons to move the severity up or down in the order.

4 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

Moving a severity moves the details of the severity to a different severity number. Thresholds are linked to the severity number. Therefore, if you move the details of Severity 7 to Severity 3, all thresholds using Severity 3 are in effect, using the Severity 7 settings.

To change the color associated with a severity

1 On the Monitor menu, click **Edit Severities**. The Severity Editor window appears.

2 Use the drop-down list in the **Severity** field to locate the severity that you want to change.

3 Complete any or all of the following fields on the Severity Editor window:

- **Color**
- **Flashing**
- **Intensity**

See page 80 for a description of these fields.

4 You can select another severity and change the color of that severity.

5 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

To change the sound associated with a severity

1 On the Monitor menu, click **Edit Severities**. The Severity Editor window appears.

2 Use the drop-down list in the **Severity** field to locate the severity that you want to change.

3 Complete the following fields on the Severity Editor window:

- **Sound**
- **Every**

See page 80 for a description of these fields.

4 Repeat this procedure to change the sound settings for another severity.

5 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

To change the action associated with a severity

1 On the Monitor menu, click **Edit Severities**. The Severity Editor window appears.

2 Use the drop-down list in the **Severity** field to locate the severity that you want to change.

3 In the **Action** field, type the command you want to execute. You can also click the browse button to locate an executable.

4 In the **every** field, type or select how often (in minutes) the action is to be performed. If this field is set to zero, the action is only performed once.

5 Choose one of the following options:

- To save your changes, click **OK**.
- To close the window without saving your changes, click **Cancel**.

An action is an operating system command that is performed when a threshold is reached. Operating system commands may include sending a pager message or sending an e-mail.

Using Drilldowns

Instance Monitor provides much more information than is initially visible on the main window. This information is contained in drilldowns.

You can access the drilldowns by choosing an option from the main toolbar or from the shortcut menu available when you right-click on a component.

This chapter contains the following information:

Using the Top Sessions drilldown

Click the Top Sessions button on the main toolbar to display the Top Sessions drilldown. The drilldown shows a list of all users connected to the Oracle database. When you first display the Top Sessions drilldown, the list is sorted in logical I/O request order.

From this screen you can:

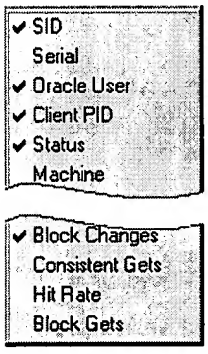
- Change the sort order and change the metric that is used to sort the list.
- Change the columns that are displayed.
- Display the shortcut menu. This menu contains links to the tabs available in the Top Sessions drilldown. It also contains two commands that can be used to trace or kill a session.

To change the sort order

1 Click on the metric or column title that you want to use to sort the list. The list is sorted in descending order by this metric.

2 To sort the metric in ascending order, click on the name of the metric again. The list is resorted in ascending order.

To change the columns that are displayed



1 Right-click on the title bar of the Top Sessions screen to display a list of available columns. The columns that are currently displayed are ticked in the list. An example of the list is shown on the left (note that the whole list is not shown in this example).

2 To select other columns, click on the name of the column. The column is included in the display.

3 To deselect a column, click on it again. The tick is removed and the column is removed from the Top Sessions screen.

The following table contains a description of the columns that you can display on the Top Sessions drilldown:

SID	Session identifier for the session
Serial	Serial number of the session. SIDs can be reused after the session disconnects, but the combination of SID and Serial number is always unique.
Oracle User	Oracle database account of the started session.
Client PID	Operating system process ID for the client program.
Status	Status of the session, ACTIVE or INACTIVE.
Machine	Name of the computer on which the client is running.
Server PID	Operating system process ID for the Oracle server process.
OS User	Operating system user for the client.
Log Reads/s	Number of logical reads per second in the past sampling interval. This includes all requests for database block, irrespective of whether they were found in the database cache.
Client pgm	The client's program.
Disk reads/s	Number of disk reads per second in the preceding sampling interval.
CPU(*100)/s	Amount of CPU utilization (in 1/100ths of a second) over the past sampling interval.
Block changes	Total number of block changes the session has performed.
Consistent gets	Total number of consistent (query) mode reads since the session was established.
Hit rate	Percentage of read requests that did not require disk I/O.
Block Gets	Total number of current (update) mode reads since the session was established.

To display the shortcut menu

From the Top Sessions drilldown:

1 Right-click on a session. The Top Sessions pop-up appears:

2 Click the option you want to select.

Session Details tab

The **Session Details** tab shows summary information for a session. The following information is shown:

Oracle SID	The session identifier.
Username	Oracle account name of the session.
Server	Oracle server program that supports this session. This may be a dedicated server, shared server, or dispatcher.
CPU Utilization (ms)	Amount of CPU time consumed by the server process, in milliseconds.
Block Gets	Total number of current (update) mode reads since the session was established.
Consistent gets	Total number of consistent (query) mode reads since the session was established.
Block changes	Total number of block changes the session has performed.
Status	Current status of the session. This can be ACTIVE or INACTIVE.
Program	Client program that initiated the session. If the full path of the program does not fit in the display area, the value in this field starts with an ellipses (...). Position the cursor over the field to see the full path in a Tooltip.
Waiting For	Details of any resource wait. See *Dealing with wait events* in the Instance Monitor Tuning Guide for more information. If the wait is for a file IO operation (such as DB FILE SEQUENTIAL READ or DB FILE SCATTERED READ) Instance Monitor displays the name of the table or index to which the IO is occurring. If you selected fast initialization when you connected to the database, the name of the table or index is not shown.
Current SQL	Current SQL statement for the session. Right-click on the SQL to display a short cut menu. You can: • Click on **Explain Plan** to display the Explain Plan window. • Click on SQLab to display SQLab Xpert.
Serial	The serial number for the session. Although the Oracle SID may be reissued as sessions connect and disconnect, the combination of SID and serial number remains unique.
OS User	Operating system user that established the session.
Client PID	Operating system process identifier for the client process that created the session.
Server PID	Process identifier of the Oracle server process that supports this session.
Phys reads	Number of physical reads initiated by the session.

Hit Rate	Percentage of block get requests that were satisfied by blocks cached in the buffer cache.
Resource Waits chart	Shows a breakdown of total resource wait categories.

Session Waits tab

The **Session Waits** tab shows the amount of time a session spent waiting for resources to become available.

When an Oracle session needs a resource that is not available or performs disk I/O, Oracle records the type and duration of the wait. The **Session Waits** tab shows the wait events in the following formats:

- The **Session Waits** panel (the top half of the tab area) shows details of each wait event. If you select the **Show Uninteresting Waits** check box (at the bottom of the tab), idle waits are included in this list.
- The **Resource Usage** chart shows a breakdown of total resource wait categories.
- The **Wait Events** chart shows a history of wait events.

Information about the current wait is shown at the bottom of the tab area. If the wait is for a file IO operation (such as DB FILE SEQUENTIAL READ or DB FILE SCATTERED READ) Instance Monitor displays the name of the table or index to which the IO is occurring.

If you selected fast initialization when you connected to the database, the name of the table or index is not shown.

For more information about wait events see *Dealing with wait events* in the Instance Monitor Tuning Guide.

Session SQL tab

The **Session SQL** tab shows the SQL in all open cursors for the session. From this screen you can use Explain Plan to explain the SQL or use SQLab Xpert to provide context-sensitive tuning advice for SQL statements. Follow these steps:

1 Right-click to display the shortcut menu.

2 You can:

- Click on **Explain SQL**. The Explain Plan window appears. See the Instance Monitor Tuning Guide for more information about the Explain Plan.

- Click on SQLab to display SQLab Xpert. See the SQLab Xpert documentation for more information on this product.

Session Locks tab

The **Session Locks** tab shows the status of open locks, locks that are pending, and objects currently accessed by the session. The tab is divided into two panels:

The left panel shows held and waiting locks. The columns in this panel are: • Type of lock. • Mode in which the lock is requested. • Mode in which the lock is held. • Is this lock blocking other users? • Object upon which the lock is held.	The right panel shows objects that are currently accessed by the session. This includes stored programs as well as segments.

Track Sessions tab

The **Track Session** tab allows you to track a session's resource usage. The following charts are provided:

- The **Wait events** chart shows the rate at which waits for various Oracle resources are occurring. For more information about wait events see *Dealing with wait events* in the Instance Monitor Tuning Guide.
- The **I/O History** chart shows the rate at which the session is performing database read I/O. The following information is displayed:

  Consistent The rate of logical IO performed for query purposes.

  Physical Indicates the rate of physical disk reads.

  Current Gets obtained for update purposes.
- The **CPU** chart shows the rate at which CPU is consumed (in $1/100^{th}$ of a second).
- The **Call History** chart shows the rate of database calls per second.

Session Statistics tab

The **Session Statistics** tab shows the value and rate of change for all Oracle statistics contained in the V$SESSTAT virtual table. These statistics are documented in the *Oracle Server Reference Manual.*

To end a session

1 Select a session in the Top Sessions drilldown.

2 Right click to display the Top Sessions shortcut menu.

3 Select **Kill session**.

4 A message appears. The message contains the session identifier and serial number of the session to be killed. Choose one of the following options:

- To kill the session, click **OK**.
- To close the message and leave the session running, click **Cancel**.

Not all users have permission to kill sessions. This permission is granted by the Database Administrator. It can be set when the user is created using the Instance Monitor User Wizard.

To trace a session

1 Select a session in the Top Sessions drilldown.

2 Right click to display the Top Sessions menu.

3 Select **Trace session**.

This is the equivalent of issuing the following command:

```
ALTER SESSIONS SET SQL_TRACE TRUE
```

For more information see your Database Administrator.

Using the Top SQL drilldown

Click the Top SQL button on the main toolbar to display the **Top SQL** drilldown. The **Criteria** tab is selected.

The screen contains the following tabs:

- Criteria tab (see below)
- SQL tab see page 93.

Criteria tab

The first panel on the Top SQL drilldown is the **Criteria** tab. You can use this panel to select and sort the SQL that is to be extracted from the shared pool.

To select and then sort the SQL

1 Type the selection criteria in the **Selection criteria** section of the tab.

Parsing user	The username of the user who first parsed the SQL statement. The statement may have been executed by other users.
Minimum executions	The number of times the SQL statement has been executed.
Minimum buffer gets	The number of logical reads the SQL statement has invoked.
Minimum disk reads	The number of physical disk reads the SQL statement has invoked.
Minimum rows	The number of rows the SQL statement has retrieved.
Contains	A text string that the SQL statement must include. Only the first 1000 bytes of SQL statements are checked.
Minimum buffer gets/executions	The average number of buffer gets per execution.
Minimum disk reads/executions	The average number of disk reads per execution.
Minimum rows/ executions	The average number of rows retrieved per execution.

Any SQL that matches all of the criteria you type is displayed. You can click the **Clear criteria** button to clear all input fields and reset the parsing user to (All users).

2 Choose which field is to be used to sort the result in the **Sorting** section of the tab.

Use this criteria...	To sort by...
Executions	Number of executions.
Disk reads	Number of disk reads.
Buffer gets	Number of logical reads (buffer gets).
Rows processed	Number of rows processed.
Buffer gets/executions	Sort by the ratio of buffer gets to executions.
Disk reads/executions	Sort by the ratio of disk reads to executions.
Rows processed/ executions	Sort by the ratio of rows processed to executions.

3 Select the sort order. You can choose **Ascending** (for example, 1 to 99 or A to Z) or **Descending** (for example, 99 to 1 or Z to A).

4 Click the **Fetch SQL** button to extract matching SQL. The SQL tab appears when the fetch operation is complete.

SQL tab

When you have typed the SQL selection criteria on the **Criteria** tab, click **Fetch SQL** to display the **SQL** tab. The SQL is displayed in the order specified on the **Criteria** tab. Click on any SQL to display the **SQL Detail** panel.

The following information is displayed on the SQL tab:

This column...	Shows...
Execs	Number of times this SQL statement has been executed.
Rows	Total number of rows retrieved by this SQL statement.
Buffer gets	Number of logical reads (buffer gets) retrieved by this SQL statement
% of total	Number of logical reads recorded for this SQL statement as a percentage of logical reads recorded for all SQL statements.
Disk reads	Number of physical disk reads recorded against this SQL statement.
Gets/Exec	Average number of logical reads per execution.
Disk/Exec	Average number of disk reads per execution.

Row/Exec	Average number of rows retrieved per execution.
User	The name of the user who first parsed this SQL statement
First Parsed	Date and time this statement was first parsed.
SQL Text	The text of the SQL statement. Some of the SQL statement might be hidden so that important sections are visible. For example, the list of columns in the SELECT list may be hidden so that tables in the FROM clause are visible.

To display the SQL Detail panel

Click on an SQL statement in the **SQL** tab. This panel shows the full SQL text for the SQL statement, and some metrics that are not shown on the **SQL** tab.

From this screen you can:

1 Right-click in the **SQL** text box. The shortcut menu displays.

2 You can:

- Click on **Explain SQL**. The Explain Plan window appears. See the Instance Monitor Tuning Guide for more information about the Explain Plan.
- Click on SQLab to display SQLab Xpert. See the SQLab Xpert documentation for more information on this product.

If the SQL cannot be explained the Explain Plan and SQLab options may be disabled.

Using the Activity drilldown

Click the **Activity** button on the main toolbar to display the Activity drilldown. The **Activity Summary** tab is selected.

The following tabs are available from this drilldown:

- Activity summary (see below)
- Wait activity (see page 97)
- Lock activity (see page 97)
- Latch activity (see page 98)
- Server activity (see page 98).

Activity Summary tab

The Activity Summary tab shows an overview of the activity on the database you are monitoring.

This tab contains the following charts:

- Logical I/O chart
- Physical I/O chart
- CPU and event waits chart
- Sessions chart
- Call rates chart
- Miss rates chart

Logical I/O chart

The **Logical I/O** chart shows the rate of logical database reads and writes. The chart shows the following details:

Block changes	The number of database blocks in memory (SGA) that are modified per second.
Current reads	The number of blocks in memory that are read in current mode per second. Current mode reads are usually associated with updates.
Consistent reads	The number of blocks in memory that are read in consistent mode. Consistent mode reads are usually associated with query activity.

Physical I/O chart

The Physical I/O chart shows I/O rates for datafile reads, datafile writes, and redo log writes. All rates are in disk I/O per second.

CPU and event waits chart

The **CPU and event waits** chart shows the amount of time (in milliseconds) that sessions have spent waiting on various events. Event wait categories from V$SYSTEM_EVENT are aggregated into categories so that they can be graphed.

See the **Wait activity** tab (on page 97) for details of specific wait events. For more information about wait events see *Dealing with wait events* in the Instance Monitor Tuning Guide.

Sessions chart

The **Sessions** chart shows the number of sessions connected to the database. The following types of sessions exist:

Idle Sessions that are not currently engaged in database activity.

Active Sessions that are currently performing some database activity.

System Sessions that are associated with Oracle background processes (such as SMON and PMON).

Call rates chart

The **Call rates** chart shows the rate of database calls made to all connected sessions. The calls shown are **Parse** (a request to prepare an SQL statement for execution), **Execute** (a statement or prepares a query to receive rows), **Commit**, and **Rollback**.

Miss rates chart

The **Miss rates** chart shows the proportion of failed attempts to acquire a resource or find a matching resource within a cache. The following miss rates are displayed:

Buffer cache The proportion of read requests that were not satisfied by a data block already cached within the SGA.

SQL area The proportion of parse requests that were not satisfied by SQL statements stored within the shared pool.

Latch The proportion of latch-get requests that were not satisfied on the first attempt.

For more information about miss rates see:

- **Buffer cache miss** alarm on page 110
- *Reducing parse overhead* and *Relieving latch contention* in the Instance Monitor Tuning Guide.

Wait activity tab

The **Wait Activity** tab is displayed by clicking the **Activity** button on the toolbar and selecting the tab. The tab shows wait events for the Oracle database (see *Dealing with wait events* in the Instance Monitor Tuning Guide).

The top half of the screen shows each individual wait event, together with the rate of waits, and time waited over the past time period.

The lower half of the screen shows the **CPU and event waits** chart (see page 96) and the **Buffer busy** chart.

Buffer busy chart

The **Buffer busy** chart shows the breakdown of buffer busy waits by buffer type. Buffer busy waits occur when a session wants to access a data block buffer in the SGA buffer cache but cannot do so because the buffer is held by another session in an incompatible mode. This usually occurs because the buffer is being modified by the other session, or because the buffer is being read in from disk by the other session.

The most common causes of buffer busy waits are:

- Concurrent insert activity into a table which has too few freelists (perhaps only one). This shows up as waits for data block (not as waits for freelist). The contention occurs because the multiple sessions are selecting the same block from a freelist and then concurrently trying to insert into that block.
- Too few rollback segments. This shows up as waits for undo header.

Lock activity tab

The **Lock Activity** tab is displayed by clicking the **Activity** button on the toolbar and selecting the tab. The tab shows a lock tree of sessions that are blocked (waiting for lock requests) and the processes that are blocking them.

The screen shows a lock tree. At the root of the lock tree are processes that hold locks but which are not waiting for locks. Processes that are waiting for locks are shown underneath the blocking processes.

You can:

- Click on a session ID (SID) to display the **Session details** and other Top Sessions tabs (see page 87) in the bottom half of the screen.
- Right-click a session to display a shortcut menu. The menu contains the **Kill Session** (see page 91) and **Trace Session** options (see page 91).

You cannot end a session unless you have kill session privileges on the database you are monitoring. For more information contact your DBA.

For information about minimizing locks, see *Reducing lock contention* in the Instance Monitor Tuning Guide.

Latch activity tab

The **Latch activity** tab is displayed by clicking the **Activity** button on the toolbar and selecting the **Latch activity** tab. For more information about latches see *Relieving lock contention* in the Instance Monitor Tuning Guide.

The top half of the screen shows absolute get, miss, and sleep rates for each latch in the instance. You can scroll up and down the list to view all of the latches.

The lower half of the screen shows the **Latch rate** chart and the **Latch efficiency** chart.

Latch rate chart

The **Latch rate** chart shows the get, miss, and sleep rate for various categories of latch. You can use the drop-down list to display get, miss, or sleep rates.

Latch efficiency chart

The **Latch efficiency** chart shows the absolute latch miss and sleep rates (as proportions of the get rate) and the percentage of non-idle event wait times that are due to latch sleeps.

Server activity tab

The **Server activity** tab is displayed by clicking the **Activity** button on the toolbar and selecting the tab. Various Oracle processes perform work on behalf of multiple client processes. This tab displays information about these processes. The information is contained in the following charts:

- Parallel query servers chart
- Servers percent busy chart
- Shared MTS servers chart
- Job processes chart

Parallel query servers chart

The **Parallel query servers** chart shows the number of busy and idle parallel query servers. Parallel query servers support parallel execution of queries and (in Oracle8) DML statements. The number of servers varies, depending on the load between the configuration parameters PARALLEL_MIN_SERVERS and PARALLEL_MAX_SERVERS.

For more information about parallel query servers see:

- **Parallel query server** alarm (on page 121)
- Server Processes panel (on page 13).

Servers percent busy chart

The **Servers percent busy** chart shows the percentage of time servers have spent in busy state. Separate graphs are shown for different types of servers. The servers that are shown are:

- Shared servers
- Dispatchers
- Parallel servers.

Shared MTS servers chart

The **Shared MTS servers** chart shows the number of busy and idle multi-threaded servers (MTS) and the number of queued SQL requests that are waiting for a server. MTS servers perform work on behalf of more than one client process. The number of shared servers varies depending on load between the values of the configuration parameters MTS_SERVERS and MTS_MAX_SERVERS.

For more information about MTS servers see:

- **Multi-threaded server** alarm (on page 120).
- *Dealing with MTS contention* in the Instance Monitor Tuning Guide.
- Server Processes panel on page 13.

Job processes chart

The **Job processes** chart shows the number of active and idle job queue processes and the number of jobs that are awaiting a free job queue (jobs pending).

For more information see **Server Processes** panel on page 13.

Using the IO drilldown

Click the I/O button on the main toolbar to display the **I/O** screen. The **I/O Summary** tab is selected.

The following tabs are available from the **I/O** screen:

- I/O Summary tab
- I/O by datafile tab
- Logical I/O tab

I/O Summary tab

The **I/O Summary** tab shows information about current I/O activity. The tab displays the following graphs:

- **I/O Rates** show the number of datafile reads per second, datafile writes per second and redo log writes per second.
- **I/O Times** shows the amount of time spent by all processes that are performing datafile reads, datafile writes, and redo writes in each second.
- **I/O by Tablespace** shows current I/O activity by tablespace. This graph can be displayed as I/O Rate (for example, I/O per second), average I/O time (average milliseconds per read), and total I/O time/second (total time in milliseconds spent reading from this tablespace).

I/O by datafile tab

The I/O by datafile tab displays I/O on a per datafile basis. This can help to identify overused (hot) datafiles. To display this tab, select the IO button from the toolbar and select this tab.

The tab contains three graphs. You can choose which graph to display by selecting an option from the drop-down list. The options are:

This option...	Shows...
I/O rate by datafile	The absolute number of I/O per second to each datafile.
Average I/O time by datafile	The average time taken to complete a read against the specified datafile.

This option...	Shows...
Total I/O time per second by datafile	The total amount of time spent reading from the datafile per second across all processes.

For more information about improving the performance of your database see *Improving database I/O* in the Instance Monitor Tuning Guide.

Logical I/O tab

The **Logical I/O** tab shows various indicators of logical or non-physical I/O requests.

Four charts are displayed on this tab. They are:

1. The **Logical I/O** chart. This is the same chart as that shown on the **Activity Summary** panel. See *Logical I/O chart* on page 95.
2. **Rollback segment logical I/O** shows the number of rollback segment accesses per second. Access are categorized as writes, reads or waits. High levels of waits may indicate a need to add more rollback segments.
3. **Row access types** breaks logical reads down into those resulting from indexed or ROWID access, and those resulting from table scans. For OLTP applications, the overhead from full table scans should generally be low.
4. **Redo generation** shows the size of redo log entries being generated per second.

Using the Alarm Log drilldown

Click the **Alarm Log** button on the main toolbar to display the **Alarm Log** drilldown. This drilldown shows the alarms that have been issued and cancelled.

An alarm is logged when an event reaches a minimum severity. The minimum severity is defined on the **Alarms** tab of the Options window. See page 48 for more information.

If the event changes to a higher severity, another alarm is logged. If it changes to a severity that is lower than the minimum severity defined on the **Alarms** tab, the alarm is cancelled.

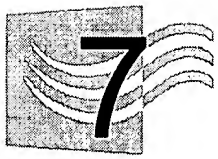
The **Alarm Log** drilldown contains all of the entries that are included in the alarm log file (defined on the **Alarms** tab of the Options window). If you chose to reuse the same alarm log file, a large number of alarms may be shown on this drilldown.

The **Alarm Log** drilldown contains the following column:

Time	The date and time the alarm was logged.
!	The following symbols are displayed in this column: Indicates this is the first time this alarm has displayed. The alarm has gone to a higher severity. The alarm has gone to a lower severity. The alarm has been cancelled.
Component	The name of the component that issued the alarm.
Action	A description of the alarm.
Details	The description from the threshold that is using the severity.
Help	If help is available the symbol appears in this column. Click on this symbol to display the help for this alarm.

To sort the alarms

1. Click the name of the column you want to sort by. The log is sorted in ascending order by this column.
2. Click the name again to sort the log in descending order by this column.
3. To sort by another column, click on the name of that column.



Responding to Instance Monitor Alarms

Instance Monitor alarms you to problems with your database by issuing an alarm. An alarm normally involves one of the following actions:

- Changing the color of the component that is experiencing the problem.
- Flashing the component.
- Sounding an audible alarm.

You can set other alarm options in the thresholds of a metric. You can also set when an alarm is displayed by changing the thresholds for the metric. See *Understanding metrics, thresholds, and severities* on page 10 for more information.

This chapter describes the alarms that may be displayed while you are using Instance Monitor and the action you should take. The following alarms are described:

Active session alarm

The **Active session** alarm becomes active when the percentage of sessions that are actively working in the database exceeds a threshold. This might not always be a serious problem, but is often a sign of a database bottleneck or heavy load.

When this alarm is current you should look at:

- Other alarms on the main Instance Monitor window. This may indicate why the number of active sessions is high.
- The **Top Sessions** page to determine which sessions are active and why.
- The **Activity** page to display which resources are most heavily waited on.

Average redo write time alarm

The **Average redo write time** alarm is activated when the time taken to write redo log entries exceeds a threshold. Unlike most other Oracle write I/Os, Oracle sessions must wait for redo log writes to complete before they can continue processing. Therefore redo log devices should be placed on fast devices.

Most modern disks should be able to process a redo log write in less than 20 milliseconds, and often much lower. To reduce redo write time see *Improving redo log writer performance* in the Instance Monitor Tuning Guide.

Buffer busy wait alarm

The **Buffer busy wait** alarm occurs when a session cannot access a block because it is in use by another session. The two most common causes are insufficient free lists for a table or insufficient rollback segments.

If the **Buffer busy wait** alarm is current, you should look at the **Wait activity** tab on the **Activity** page. This tab shows you the actual number and duration of buffer busy waits, as well as a breakdown of buffer busy waits by buffer type.

If the predominant buffer waits are for data blocks, it is likely that you need to create multiple free lists (using the FREELIST clause of the CREATE TABLE statement) for tables that are subject to very heavy concurrent inserts. Waits for data blocks occur when multiple sessions select the same block from a freelist and concurrently try to insert into that block.

If the leading category is for either undo header or undo block you may need to create additional rollback segments.

Buffer cache miss alarm

The **Buffer cache miss** alarm occurs when the buffer cache miss rate exceeds a threshold.

To see the exact miss rate, look at the **Miss rates** chart on the **Activity** drilldown. This ratio describes the percentage of time a required data block was not found in the buffer cache. For most applications, a value under 10% (and often much lower) is desirable.

The buffer cache hit ratio is one of the most significant tuning ratios. Untuned values can lead to unnecessarily high disk I/O rates and contention for internal resources (latches).

To improve the buffer cache hit ratio, you can increase the size of the buffer cache by increasing the size of the DB_BLOCK_BUFFERS configuration parameter.

Applications that perform frequent table scans of large tables (such as data warehouses) may see little benefit from increasing the buffer cache. For these applications, low buffer cache hit ratios may be unavoidable. You may want to adjust your thresholds for this alarm to suit your database. For information on changing a threshold see *Managing thresholds* on page 69.

Cache buffer chains latch alarm

The **Cache buffer chains latch** alarm occurs because contention for the cache buffer chains latch exceeds a threshold.

Two main latches protect data blocks in the buffer cache:

- The cache buffer LRU chain latch must be obtained in order to introduce a new block into the buffer cache and when writing a buffer back to disk.
- The cache buffer chains latch is acquired whenever a block in the buffer cache is accessed (pinned).

Contention for these latches usually occurs in a database that has very high I/O rates. It may be possible to reduce contention for the cache buffer LRU chain latch by increasing the size of the buffer cache. Doing this reduces the rate at which new blocks are introduced into the buffer cache.

Reducing contention for the cache buffer chains latch usually requires reducing logical I/O rates by tuning and minimizing the I/O requirements of application SQL.

You can create additional cache buffer LRU chain latches by adjusting the configuration parameter DB_BLOCK_LRU_LATCHES. You may be able to reduce the load on the cache buffer chain latches by increasing the configuration parameter DB_BLOCK_HASH_BUCKETS.

Cache buffer LRU chains latch alarm

The **Cache buffer LRU chains latch** alarm occurs because contention for the cache buffer chains latch exceeds a threshold.

Two main latches protect data blocks in the buffer cache. They are:

- The cache buffer LRU chain latch must be obtained in order to introduce a new block into the buffer cache, and when writing a buffer back to disk.
- A cache buffer chains latch is acquired whenever a block in the buffer cache is accessed (pinned).

Contention for these latches usually occurs in a database that has very high I/O rates. It may be possible to reduce contention for the cache buffer LRU chain latch by increasing the size of the buffer cache. Doing so reduces the rate at which new blocks are introduced into the buffer cache.

Reducing contention for the cache buffer chains latch usually requires the reduction of I/O rates by tuning and minimizing the I/O requirements of application SQL.

You can create additional cache buffer LRU chain latches by adjusting the configuration parameter DB_BLOCK_LRU_LATCHES. You may be able to reduce load on the cache buffer chain latches by increasing the configuration parameter DB_BLOCK_HASH_BUCKETS.

Datafile read time alarm

The **Datafile read time** alarm occurs if the average time for a random datafile read exceeds a given threshold. Most modern disk devices should be able to satisfy a random I/O request in less than 20 milliseconds. Values that are much higher than this may indicate that there is contention for disk devices. For more information see *Improving database I/O* in the Instance Monitor Tuning Guide.

Extent failure alarm

The **Extent failure** alarm occurs when Instance Monitor detects that a segment within the database may fail to allocate its next extent.

You should use a space management tool (such as SQLab Space Manager) to identify and correct segment storage. You can also identify the segments that may fail to extend by issuing the following statement:

```
select owner,segment_name, next_extent,
s.tablespace_name,max_free_bytes
  from sys.dba_segments s,
       (select tablespace_name,max(bytes) max_free_bytes
          from sys.dba_free_space
         group by tablespace_name) f
 where s.next_extent > f.max_free_bytes
   and s.tablespace_name=f.tablespace_name
```

Free buffer waits alarm

The **Free buffer waits** alarm occurs when free buffer waits (as a proportion of total waits) exceeds a threshold.

Free buffer waits occur when a session wants to read a data block from a database file on disk into the buffer cache. If there are no unmodified (or clean) blocks in the buffer cache, the session has to wait for the Database Writer process to write modified (or dirty) blocks to disk in order for free buffers to be made available. Normally, the Database Writer is constantly writing dirty buffers to disk, so this event should rarely occur. When it does occur it is usually due to one of the following reasons:

- Untuned disk layout.

  If datafiles are not spread evenly across disk devices then a single disk may form a bottleneck to both read and write performance. When this happens, the Database Writer may not be able to clear dirty blocks from this device as rapidly as they are created.

- Untuned Database Writers.

  To efficiently write to multiple disk devices it is essential that you either configure multiple Database Writers or implement asynchronous or list I/O. This helps the Database Writer keep up with changes to the buffer cache.

- Untuned sorts.

  Prior to Oracle 7.2, large sorts requiring a temporary segment would write the sort blocks to the buffer cache and rely on the Database Writer to move them into the temporary segment. This often flooded the buffer cache and caused other sessions to encounter free buffer waits. In Oracle 7.2 this can be avoided by setting the configuration parameter SORT_DIRECT_WRITE to True. This enables sorts to avoid the buffer cache completely. In Oracle 7.3 and above, the sort direct writes are enabled by default in most circumstances.

For more information see *Improving database writer performance* in the Instance Monitor Tuning Guide.

Library cache miss ratio alarm

The **Library cache miss ratio** alarm occurs when the library cache hit ratio falls below a given threshold.

The library cache hit ratio describes the frequency with which a matching SQL statement is found in the shared pool when an SQL parse request is issued by a session. If a matching SQL statement is not found in the library cache, the SQL statement must be parsed and loaded into the library cache. Low hit rates therefore result in high CPU consumption (from parsing) and possible contention for library cache latches (when the new SQL is loaded into the library cache). An acceptable rate for the library cache get hit rate is 90-95% or higher.

The most frequent cause of high miss rates in the library cache is the use of literals rather than bind variables in SQL statements. Bind variables reduce parse overhead by allowing otherwise identical SQL statements with different query parameters to be matched in the shared pool. Bind variables, however, preclude the use of column histograms and are therefore not suitable in all circumstances.

Lock wait alarm

The **Lock wait** alarm occurs when the proportion of time sessions spend waiting for locks exceeds a threshold.

Lock waits can occur for a number of reasons, sometimes including internal database contention. However, the most common category of locks are those taken out on table rows or index entries during an update, insert, or delete.

For more information see:

- The **Lock activity** tab (on page 97) for details of all locks on the system.
- *Reducing lock contention* in the Instance Monitor Tuning Guide for information about reducing lock overhead.

Log buffer space wait alarm

The **Log buffer space wait** alarm occurs if the amount of time spent waiting for space in the redo log buffer exceeds a threshold. The redo buffer improves session performance by buffering redo log changes in memory. However, if the redo buffer is too small, or if the redo log writer is too slow, then sessions may spend time waiting for space in the redo buffer to be made available.

Increasing the size of the initialization parameter LOG_BUFFER can help. However, if the real problem is the speed of the redo log writer process, then consider the recommendations in *Improving redo log writer performance* in the Instance Monitor Tuning Guide.

Log switch time alarm

The **Log switch time** alarm occurs if the amount of time spent waiting for log switch events exceeds a threshold.

A log switch must occur whenever a log file is full. Usually, the log switch time is trivial. However, it can become significant in the following circumstances:

- The next redo log to be used is not available because the checkpoint that started when the log was switched is not complete.
- The redo log has not yet been archived.

These problems usually occur when redo log I/O rates are sub-optimal, or when the number or size of redo logs is insufficient. See *Improving redo log writer performance* in the Instance Monitor Tuning Guide.

Multi-threaded server alarm

The **Multi-threaded server** alarm occurs when all or most multi-threaded server (MTS) or dispatcher processes are busy.

In an MTS environment, multiple sessions share a smaller number of shared server processes. These processes perform database activities on their behalf. The servers are usually allocated by a smaller number of dispatcher processes.

The number of servers and dispatchers can increase up to the limits set by the configuration parameters MTS_MAX_DISPATCHERS and MTS_MAX_SERVERS. If the demand for the processes causes all servers to become busy, sessions may experience long wait times before a server becomes available.

See the **Server activity** tab on the Activity drilldown to obtain the precise load on MTS servers and dispatchers. If sessions are waiting on servers, either increase the number of servers or use dedicated server processes.

Parallel query server alarm

Parallel server processes perform work to resolve SQL statements that are being executed in parallel. If all parallel server processes are busy, parallel SQL may be executed in serial, increasing its execution time. The **Parallel query server** alarm occurs when all or most parallel servers are busy.

You can increase the number of parallel servers by increasing the value of the initialization parameter PARALLEL_MAX_SERVERS. Increasing this parameter can lead to reduced performance if the number of servers active exceeds the capacity of your CPU or disk subsystems.

For more information see the **Parallel query servers** chart (on page 99) and **Server Processes** panel (on page 13).

Parse ratio alarm

The **Parse ratio** alarm is raised when the ratio of parse calls to execute calls exceeds a threshold. A parse call is required to prepare an SQL statement for execution. Once parsed, a statement can be executed many times. Excessive parsing wastes CPU and can cause latch contention.

For more information see *Reducing parse overhead* in the Instance Monitor Tuning Guide.

Redo allocation and copy latch alarms

Two latches control access to the redo buffer. The redo allocation latch must be acquired in order to allocate space within the buffer. If the redo log entry to be made is greater than the configuration parameter LOG_ENTRY_MAX_SIZE, the session that acquires the redo allocation latch may copy the entry into the redo buffer immediately while holding the allocation latch.

If the log entry is greater than LOG_ENTRY_MAX_SIZE, the session releases the redo allocation latch and acquires the redo copy latch in order to copy the entry.

There is only one redo allocation latch, but there can be multiple allocation latches. The number of allocation latches is set in the LOG_SIMULTANEOUS_COPIES parameter.

If you see contention for the redo allocation latch, reduce the number of redo buffer copies that occur on this latch by decreasing LOG_ENTRY_MAX_SIZE. If you see contention for the redo copy latch, increase the number of copy latches by increasing the value of LOG_SIMULTANEOUS_COPIES.

SQL cache miss rate alarm

The **SQL cache miss rate** alarm indicates that SQL statements being submitted are not finding matching statements in the SHARED_POOL. The most likely causes are:

- Failure to use bind variables. See *Reducing parse overhead* in the Instance Monitor Tuning Guide.
- SHARED_POOL is too small.

Unarchived logs alarm

The **Unarchived logs** alarm occurs if the number of unarchived redo logs exceeds a threshold. If all on-line logs need archiving then database update activity can be suspended while the redo log writer waits for the next log to be archived.

For information on improving redo log write performance, see *Improving redo log writer performance* in the Instance Monitor Tuning Guide.

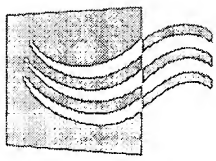
Write complete wait alarm

The **Write complete wait** alarm occurs when write complete waits (as a proportion of total waits) exceeds a threshold.

Write complete waits occur when a session tries to modify a block that is currently being written to disk by the Database Writer process. This happens occasionally (especially during checkpoints) but if it is contributing significantly to overall waits then it may indicate inefficiency in the Database Writer. The solution may involve optimizing datafile I/O and Database Writer configuration by

- Spreading datafiles across multiple disks
- Using multiple Database Writers
- Employing asynchronous or list I/O.

For more information, see *Improving database writer performance* in the Instance Monitor Tuning Guide.



Glossary

Activity button

Click the Activity button to display the activity drilldowns. These include:

- Activity Summary tab
- Wait activity tab
- Lock activity tab
- Latch activity tab
- Server activity tab.

aggregate operations

Operations that group related rows and return a single row for each group. For example, returning the total number of employees in each department. Aggregate operations are invoked with the group by operator.

Alarm Log button

Click the Alarm Log button to display the Alarm log. The Alarm log contains information about the alarms that have been raised in this Instance Monitor session.

archiver process (ARCH)

This Oracle process copies completed redo logs to backup storage.

array processing

Allows a single SQL call to process multiple rows. For example, a single execution of an insert statement could add multiple rows, or a single fetch from a select statement could return multiple rows.

In programming environments, array variables are used to hold the rows fetched or inserted. In many development and inquiry tools, array processing is enabled transparently and automatically.

artificial key

A unique key that contains no real-world information. Artificial keys are usually generated using Oracle sequences. Compare with natural key.

asynchronous An asynchronous call is one that can return control before the operation has completed. Asynchronous IO allows a process to queue requests to multiple devices concurrently. Asynchronous IO allows a process to submit multiple IO requests without waiting for each request to complete. In practice, this means a single process can utilize the bandwidth of multiple disks.

B*-tree index An index structure which takes the form of a hierarchy or inverted tree. This is the default format for Oracle indexes.

background processes Perform specialized tasks on behalf of all sessions. For example, the Database Writer (DBWR) is responsible for writing changed blocks from the buffer cache to the database files. The log writer (LGWR) is responsible for writing blocks from the redo buffer to the redo logs. The archiver process (ARCH) copies completed redo logs to backup storage. Other processes (such as SMON and PMON) perform housekeeping functions, and some processes can only be enabled if certain Oracle options are enabled.

binary chop A procedure for searching a sorted list of items. The list is successively divided into two sections and the section that must contain the desired item further sub-divided. Eventually the remaining portion is sufficiently small to enable a sequential scan. This technique is useful in programs that cache table data to avoid excessive database access.

bind variables Allow the variable portions of an SQL statement, such as data values, to be inserted or a search key, to be defined as "parameters" to the SQL statement.

The use of bind variables allows SQL statements to be re-executed without re-parsing the SQL statement. The alternative approach, where substitution variables are embedded as literals within the SQL statement requires that the SQL statement be re-parsed when re-executed.

block The basic unit of storage in an Oracle instance. Block sizes most commonly range between 2 and 8 KB.

branch blocks The middle level of blocks in a B-tree index. Each branch block contains a range of index key values and pointers to the appropriate leaf blocks.

browse button

Click the **Browse** button to search for a file or to specify the drive and directory where a file is to be saved.

When you click this button, the Open window appears. Use the standard Windows commands to locate the file or directory.

When you have found the file or directory, click **Open**. The file and pathname are shown in the field where you clicked the **Browse** button.

buffer cache

An area in the SGA that contains copies of blocks from database files. The buffer cache exists primarily to reduce disk I/O. It does this by allowing sessions to access frequently or recently accessed data in memory.

cache buffer LRU chain

The cache buffer LRU chain latch protects the LRU list. The LRU list records how recently a block was accessed.

calibration

Determines the maximum and minimum values for every dataflow by observing the data moving through the database system. This information helps Instance Monitor draw the dataflows correctly. You can manually override these calibrated thresholds at any time, and for any given dataflow.

cardinality

A measure of the number of unique values within a column or an index. The higher the cardinality of the index, the fewer the number of rows that are returned by an exact lookup of a key value (and hence the more useful the index).

chart

A graphical representation of a statistic over a period of time. One or more statistics may be shown on the same graph.

To highlight an area of the chart, position your cursor at the top left corner of the area you want to highlight. Click and hold the left mouse button and drag it across the area you want to highlight. An outline is shown around the area you selected. When you release the mouse button, the graph is redrawn showing just the area you highlighted. You can continue to zoom into the area you have highlighted.

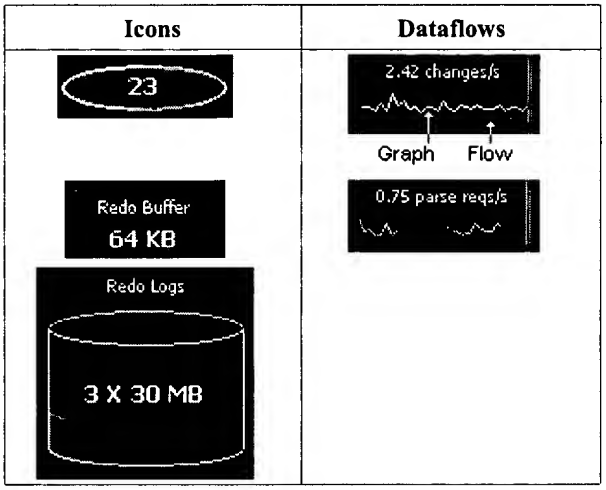
To display the graph at normal size, click and hold the mouse button in the graph. Move the cursor up and to the left. When you release the mouse button the graph is redrawn at normal size.

checkpoint The process of writing all modified blocks in the buffer cache to disk.

client A software application that requests the services, data, or processing of another application or computer (known as the server).

column A portion of a database table that stores a particular type of information. When a column is defined, it is given a name and datatype. A table of addresses, for example, might contain a column called CITY and another called TELEPHONE NUMBER.

component The icons and dataflows shown on the main Instance Monitor window. The following diagrams show examples of components.



See also *dataflow*, *icon*, and *label*.

concatenated index An index which is comprised from more than one column.

connect string The string that is used to link to a database. The database name is defined within Oracle utilities.

connectivity software A program that is used to establish and maintain a connection between a database and a client application. Once a connection is established, the client can access, modify, and store data on the database.

Under Oracle, you can use SQL*Net or Net8 (for Oracle 8) to establish a connection to a database. Under SQL Server, you can use the ODBC drivers provided with Windows.

consistent read Oracle queries return rows that are consistent with the time at which the query commenced. This consistent read may require access to rows that have changed since the query commenced - these rows are accessed from rollback segments.

cost-based optimizer Determines the execution plan based on an estimate of the computer resources (the cost) required to satisfy various access methods. The cost-based optimizer uses statistics (including the number of rows in a table and the number of distinct values in indexes) to determine the optimum plan.

cursor A memory structure that contains an SQL statement or PL/SQL block. Also referred to as a context area.

database connection A link to a database and its data. This link is established when you log on to the database.

database files Files that contain the data that makes up an Oracle database.

database object Anything that is stored in a database, including tables, views, indexes, snapshots, triggers, and stored program units.

database segments Include user objects such as tables and indexes, as well as rollback segments and temporary segments. A segment can belong to only one tablespace.

database writer (DBWR) The process that is responsible for writing changed blocks from the buffer cache to the database files.

dataflow A line graph on the main Instance Monitor window. Dataflows depict the flow of information between different components of the database management system. The color of a dataflow can change in response to the data that is displayed.

You can display a dataflow as a pulse or as a flow and a graph.

For more information see *flow*, *graph*, and *pulse*.

DBA Database Administrator. The person who maintains the databases in your organization.

dedicated server Processes that perform work on behalf of a single client process. The number of dedicated servers varies as users log in and out of the database.

denormalization The process of re-introducing redundant or derived information into a data model with the aim of improving performance.

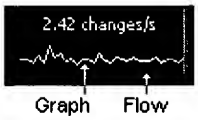
driving table The table that is accessed first in a table join. Choosing the best driving table is a key decision when optimizing join order.

extent Segments are composed of a number of distinct storage allocations known as extents. As a segment grows it allocates extents as required, up to the limit of available space and the value of maxextents.

flow The flow shows you the current level of activity. As the rate of data transfer increases, so too does the speed of the flow. If the statistic represented by the flow moves into another threshold, the flow may change color. The combination of movement and color makes it easy to spot congested areas.

The graph sits on top of the flow and shows you how the load has varied over time.

The following diagram shows an example of a flow and graph.



foreign key A column or columns within one table which relate to the primary key of a master or parent table. These matching foreign and primary key columns can be used to join the two tables.

free buffer waits Waits that occur when a session wants to read a data block from a database file on disk into the buffer cache. If there are no unmodified (or clean) blocks in the buffer cache, the session has to wait for the Database Writer process to write modified (or dirty) blocks to disk in order for free buffers to be made available.

free lists A free list is a list of blocks that are eligible for insert. Each segment contains at least one freelist. Multiple freelists can be configured using the freelists clause if the segment is subject to high concurrent insert rates.

Multiple freelist groups can be configured in an Oracle Parallel Server environment so that each instance inserts rows into specific blocks.

graph

A white line that sits on top of a pulse. The graph represents how the load on the database has varied over time.

See *flow* for more information.

hash value

Hashing refers to the technique of mathematically transforming a key value into a relative address which can be used to rapidly locate record. This relative address is known as a hash value. Oracle uses hashing as a table access method (hash clusters) and to optimize certain join operations (hash join). Hashing is also used extensively within internal SGA operations.

hashing

In general, hashing refers to the technique of mathematically transforming a key value into a relative address that can be used to rapidly locate records. Oracle uses hashing as a table access method (hash clusters) and to optimize certain join operations (hash join). Hashing is also used extensively within internal SGA operations.

hierarchical queries

A special case of a self-join in which each row accessed child rows in a hierarchy of parent-child relationships. This is sometimes referred to as explosion-of-parts.

high water mark

The highest block in a segment that has ever contained data. The high water mark increases as rows are inserted into the segment. Deleting rows does not reduce the high water mark.

Full table scans access all rows in the segment up to the high water mark.

I/O

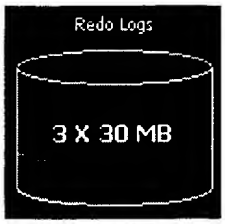
Input or output to a peripheral device. In an Oracle context, I/O refers to input or output disk devices.

icon

The icons in Instance Monitor fall into the following categories:

Process icons are oval in shape and contain a single value that represents the state or existence of a database process.

Memory icons are rectangular and show the utilization of database-specific areas in memory.



Disk icons are cylindrical and fill up as a file increases in size.

Meters show a measurement. The highest and lowest possible values of the measurement are shown.

IO button

Click the IO button to display the I/O drilldowns. These include:

- I/O Summary tab
- I/O by datafile tab
- Logical I/O tab.

label

Labels are shown above most icons and dataflows.

A label may have different metrics and thresholds to the component it is over. You can also tailor the metrics and thresholds of the labels.

latch

An internal Oracle locking mechanisms. Latches prevent multiple sessions from simultaneously updating the same item within Oracle shared memory (SGA).

latch free wait

The wait that occurs when a session needs to acquire a latch that is held by another session.

latch miss

If a process requires a latch and cannot obtain it on the first attempt, a latch miss results. The session repeatedly attempts to obtain the latch up to value of the configuration parameter SPIN_COUNT.

leaf blocks

The lowest level of blocks in a B-tree index. Each leaf block contains a range of index key values and pointers (ROWIDs) to appropriate blocks.

least recently used (LRU) list
The least recently used (LRU) algorithm is used by Oracle to remove cached data blocks that have least recently been accessed. When a block is read from disk, it is placed on the Most Recently Used end of the LRU list, unless it has been read in from a table scan of a large table and the CACHE hint has not been specified.

list I/O
List I/O allows multiple IO requests to be submitted in a single call. It is similar to asynchronous IO.

log file parallel write event
The log file parallel write wait event occurs when the log writer waits for IO to redo log devices to complete.

log file sync
The log file sync wait event occurs when a session issues a commit and must wait for a redo log IO before completing.

log writer (LGWR)
The Oracle process that is responsible for writing blocks from the redo buffer to the redo logs.

main menu bar
The menu bar is shown at the top of the Instance Monitor window. It contains the following options. They are:

- Monitor
- Navigator
- Tools
- Help.

metric
A unit of measurement that can be applied to a database. Metrics can help you gauge the performance of a database system.

multi-threaded servers
See *server processes*.

natural key
A unique identifier for a table that is composed of naturally occurring columns in the table. Compare with artificial key.

nested loops join
A join method in which each row of the outer table is read. For each row, a lookup of the inner table is undertaken. This best suits joins where the inner table is accessed via an index lookup.

NULL values
Indicates that a value is missing, unknown, or inapplicable. The use of null values extends the normal two-valued logic to a three-valued logic. Null values are important in SQL tuning because they are not generally stored in indexes and therefore present unique tuning problems.

OLAP — On-line Analytical Processing. Involves the real-time manipulation of large quantities of data generally for the purpose of facilitating business decisions. OLAP databases are typified by large data volumes and infrequent, long running queries.

OLTP — On-line Transaction Processing. OLTP databases typically have a very high rate of update and query activity. OLTP is typified by high rates of index lookups, single-row modifications, and frequent commits.

optimistic locking strategy — A locking strategy based on the assumption that a row is unlikely to be changed by another session between the time the row is queried and the time it is modified. Optimistic locking minimizes the lock duration but requires that the transaction be aborted if the row is changed by another session.

optimizer — The component of the Oracle software that determines the execution plan for an SQL statement. Oracle supports two approaches to query optimization. They are:

- The rule-based optimizer determines the execution plan based on a set of rules. The rules rank various access paths. For example, an index-based retrieval has a lower rank than a full table scan. A rule-based optimizer uses indexes wherever possible.
- The cost-based optimizer determines the execution plan based on an estimate of the computer resources (the cost) required to satisfy various access methods. The cost-based optimizer uses statistics (including the number of rows in a table and the number of distinct values in indexes) to determine the optimum plan.

Oracle System Global Area — See *System Global Area (SGA)*.

outer table — The first table processed in a join of two tables.

page — A connection to a database.

panel — A group of related components (normally icons) on the main Instance Monitor window. The name of the panel is normally shown at the top of the panel.

parallel execution The execution of an SQL operation using multiple processes or threads. This allows some of the stages of execution to be executed simultaneously and large tables to be scanned by multiple processes.

parsing The process of preparing a SQL statement for execution. This involves checking the statement for syntax errors, checking for a matching statement in the shared pool, and determining the optimal execution plan. Parsing can contribute significantly the processing overhead, especially in OLTP-like environments.

pessimistic locking strategy A locking strategy based on the assumption that a row might be changed between the time it is fetched and the time it is updated. Pessimistic locking involves locking the row when it is selected to prevent any concurrent updates.

pinned The process of accessing a block in the buffer cache.

positioning buttons Instance Monitor provides two buttons you can use to change the order of thresholds and severities. The buttons are:

Click this button to move the item up in the list. For example, to move it from position 3 to position 1.

The other items in the list are renumbered.

Click this button to move the item down in the list. For example, to move it from position 2 to position 6.

The other items in the list are renumbered.

primary key A column or columns that uniquely identify a row in a table.

process A unit of execution in a multi-processing environment. A process typically executes a specific program and has a unique and private allocation of memory. The operating system determines the process's access to resources such as CPU, physical memory, and disk.

pulse

The pulse moves in the direction of the dataflow. As the rate of data transfer increases, so too does the speed of the pulse.

The pulse can change color if the statistic represented by the pulse moves into another threshold.

The combination of movement and color makes it possible to identify congested areas quickly.

The following diagram shows an example of a pulse with a label:

query

A SQL statement that returns a set of values from one or more tables in the database. Instance Monitor uses a variety of queries to collect information about a database's performance.

pause button

Click the Pause button to stop Instance Monitor collecting information.

To restart the data collection, click the button again.

positioning buttons

Instance Monitor provides two buttons you can use to change the order of thresholds and severities. The buttons are:

Click this button to move the item up in the list. For example, to move it from position 3 to position 1.

The other items in the list are renumbered.

Click this button to move the item down in the list. For example, to move it from position 2 to position 6.

The other items in the list are renumbered.

RAID

Redundant Array of Inexpensive Disks. RAID is used to describe the configuration of multiple physicals disks into one or more logical disks. There are three main types of RAID. They are:

- **RAID 0**. Sometimes referred to as striping disks. In this configuration, a logical disk is constructed from multiple physical disks.
- **RAID 1**. Referred to as disk mirroring. In this configuration, a logical disk is comprised of two physical disks. In the event that one physical disk fails, processing can continue using the other physical disk.
- **RAID 5**. Stripes data across multiple drives while storing sufficient parity information on all drives to allow data to be recovered should any single drive fail.

random I/O

I/O In which a specific disk block is directly accessed. This is typical of the I/O that results from indexed lookups.

redo allocation

The redo allocation latch must be acquired before a session can allocate space in the redo log buffer.

redo buffer

Contains redo log entries that have not yet been written to the redo logs. The redo buffer is periodically flushed and is always flushed when a commit occurs.

redo copy latch

One of the redo allocation latches must be acquired before a session can copy an entry into the redo log buffer.

redo logs

Oracle files that are used to record all changes made to objects within a database. When a commit is issued, the changes made within the transaction are recorded in the redo log. The redo log can be used to restore the transaction in the event of a system failure.

referential integrity

Referential integrity ensures that foreign keys correctly map to primary keys. A referential constraint prevents the insert or update of foreign keys for which there are no matching primary keys. It either prevents the deletion of primary keys if foreign keys exist or deletes these foreign key rows (delete cascade).

Referential integrity can result in table-level locking if there are no indexes on the foreign keys.

result sets	The output from a SQL query. Results sets have the same tabular construction as tables. Results sets are also created during intermediate SQL operations. For instance in a multi-table join, each successive join creates a result set. The final result set is the output of the query.
rollback segments	Segments that store the contents of a row before it is modified by a DML (update, insert delete) statement. This information is used in the event of a rollback, to provide a consistent view of the table for queries that commenced before the transaction was committed and to record the eventual success of a transaction.
row level locking	In general, Oracle only ever locks a row that is modified by a DML statement. Page, block, or table locks are not normally applied and read locks are never applied.
ROWID	The ID that uniquely identifies a row by its physical location. The ROWID of a row (if known) is the fastest way to access a row. An index contains the ROWIDs that match specific key values, thus providing quick access to these rows.
rule-based optimizer	The rule-based optimizer determines the execution plan based on a set of rules. The rules rank various access paths. For example, an index-based retrieval has a lower rank than a full table scan. A rule-based optimizer uses indexes wherever possible.
segment	An object in an Oracle database that consumes storage. Examples are tables, indexes, rollback segments, temporary segments, and clusters.
selectivity	A measure of the number of table entries for each index key. The less rows in the table that match specific index keys, the more selective is the index.
sequence table	A sequence table is a table containing sequence number information and is an alternative to using Oracle sequence generators.
sequential I/O	I/O in which disk blocks are read in sequence. This is typical of the I/O that results from full table scans.
serial execution	The execution of an SQL statement using a single process or thread. This requires that each stage of the SQL operation be processed one after the other. Compare with parallel execution.

server processes The process that performs SQL processing on behalf of an Oracle session. A server process can perform processing for a single session only (known as a dedicated server) or for multiple sessions (known as a multi-threaded server).

See also *shadow process*.

severity Describes the level of importance of a threshold. A severity is user-defined and determines how Instance Monitor behaves when the values for a metric fall within a range of values. For example, unusually large values might force a metric into a threshold with a high severity. This in turn could change the color of a component, play a sound, or execute an operating-system command.

SGA See *System Global Area*.

shadow process In many environments, the Oracle program runs in a separate process from the client program (for instance, SQL*PLUS). This server process is referred to as the shadow process.

shared pool An area of the SGA that stores parsed SQL statements, data dictionary information, and some session information. The shared pool reduces parse overhead by caching frequently executed SQL statements.

SMP Symmetric Multi-processing. A SMP machine contains multiple, equivalent CPUs. The SMP architecture dominates mid-range UNIX computers and is increasingly popular on Microsoft NT systems.

spike An abnormally high maximum value in a dataflow or graph.

spin lock If a process requires a latch and cannot obtain it on the first attempt, a latch miss results. The session repeatedly attempts to obtain the latch up to value of the configuration parameter SPIN_COUNT. This technique is known as acquiring a spin lock.

SQL button Click the SQL button to display the SQL drilldowns.

standard deviation A measure of how widely values diverge from the mean.

striping
A familiar term for RAID 0. Striping involves spreading data evenly across a number of disks, thus allowing higher data transfer rates than would otherwise be possible.

System Global Area (SGA)
An area of shared memory that stores information that can be shared by multiple sessions.

tablespace
A logical structure that contains and groups together the segments (mainly tables and indexes) that make up a database. A tablespace can consist of more than one database file, but any given database file can belong to only one tablespace.

temporary segments
The storage space for data that is needed for large sorts or for the large, intermediate temporary tables created during SQL statement execution.

thread
A unit of execution that shares its memory space with other threads. Threads can be implemented within processes on some systems or may be used in place of processes in others (for instance, in Windows NT).

threshold
A range of values that might be returned by a metric. If the metric falls within this range, Instance Monitor checks the threshold's severity to determine how to behave. For example, the component representing the metric might change color.

Tooltip
A message that appears whenever the mouse cursor moves over certain areas of the screen.

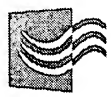
Top Sessions button

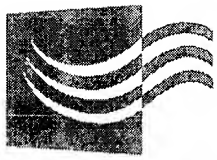
Click the Top Sessions button to display the Top Sessions drilldowns. These include:

- Session Details tab
- Session Waits tab
- Session SQL tab
- Session Locks tab
- Track Session tab
- Session Statistics tab.

transaction
A set of DML (update, delete or insert) operations that succeed or fail as a unit. A transaction is successfully terminated by the COMMIT statement or aborted with the ROLLBACK statement.

transaction slot Each database block contains a number of transaction slots, controlled by the parameters INITRANS and MAXTRANS, sometimes also referred to as Interested Transaction List (ITL) entries. A session must acquire one of these entries to acquire or wait for a row level lock.





Index

D

E

F

G

H

I

Q

R

S

T

U

W